



Escola Politècnica Superior
d'Enginyeria de Vilanova i la Geltrú

UNIVERSITAT POLITÈCNICA DE CATALUNYA

PROYECTO FINAL DE CARRERA

TÍTULO:

Diseño de un sistema de posturografía portátil

AUTOR: Anna Alonso Pastor

TITULACIÓN: ETI Electrónica Industrial

DIRECTOR: Cristóbal Raya Giner

DEPARTAMENTO: ESAII

Este proyecto tiene en cuenta aspecto medioambientales: ☐ Sí ☐ No

PROYECTO FINAL DE CARRERA

RESUMEN (máximo 50 líneas)

El objetivo del proyecto se basa en el diseño de un sistema de posturografía portátil, el cual nos permite evaluar el equilibrio de una persona mediante el cálculo de la proyección de su centro de gravedad. Esta técnica es muy utilizada en el ámbito médico para el diagnóstico y rehabilitación de los pacientes con alteraciones del equilibrio y trastornos neurosensoriales.

El sistema de posturografía estará compuesto por una plataforma con 4 sensores, una placa de circuito impreso y un programa de aplicación en el ordenador.

Para la plataforma con los 4 sensores se aprovechara la estructura de una báscula así como los sensores que contiene, cada sensor está compuesto por 2 galgas extensiométricas.

La placa de circuito impreso está formada por diferentes módulos con una función concreta. Dispondrá de un circuito de alimentación para suministrar las diferentes tensiones de alimentación que necesitan los dispositivos. Un circuito de acondicionamiento para obtener una señal asociada a la fuerza ejercida, sin offset y amplificada. Un circuito de procesamiento de datos consistente en un microcontrolador dsPIC que convertirá las señales analógicas de los 4 sensores en señales digitales, almacenará 10 valores de cada sensor, realizará la media y lo transmitirá por la UART. Y un circuito de comunicación consistente en un módulo Bluetooth que recibirá los datos de la UART y los transmitirá al ordenador.

La aplicación del ordenador recibirá los datos por Bluetooth y los tratará de manera que se muestre por pantalla un gráfico que indique en todo momento dónde se encuentra la proyección del centro de gravedad de la persona sobre la plataforma.

Palabras clave (máximo 10)

Posturografía	Equilibrio	Portátil	Galga
dsPIC	Bluetooth		

Sumario

Capítulo 1. Introducción

En este capítulo se hace una introducción en el ámbito de la posturografía y se establecen los objetivos del proyecto analizando los antecedentes y las necesidades.

Capítulo 2. Diseño del circuito de alimentación, instrumentación y acondicionamiento

En este capítulo se desarrollan los conceptos utilizados para el diseño de la parte analógica del hardware, formada por los sensores, el circuito de acondicionamiento y el circuito de alimentación.

Capítulo 3. Procesador de señal

En este capítulo hace referencia a la parte digital del circuito, y en él se definen las características del microcontrolador y se desarrollan los conceptos utilizados para su configuración según los requisitos deseados para el proyecto.

Capítulo 4. Comunicación con el PC

En este capítulo se define el sistema de comunicación empleado, así como sus características y procedimientos seguidos para su configuración.

Capítulo 5. Elaboración y diseño de la PCB

En este capítulo se define el concepto de PCB así como el software utilizado para el diseño, el método de elaboración y los resultados obtenidos.

Capítulo 6. Programas realizados

En este capítulo se hace referencia al Hardware del proyecto compuesto por el programa del microcontrolador y la aplicación de ejemplo para el ordenador.

Capítulo 7. Presupuesto

En este capítulo se incluye el presupuesto completo del proyecto.

Capítulo 8. Conclusiones

En este capítulo se definen las conclusiones obtenidas de la elaboración del proyecto y se establecen posibles líneas futuras de trabajo.

Capítulo 9. Bibliografía

En este capítulo se exponen las referencias tanto de libros, artículos, manuales como de páginas web empleadas para el desarrollo del proyecto.

Anexo I. Diseño de una PCB en Altium Designer

En este anexo se incluye un tutorial para el diseño de PCB utilizando el software Altium Designer Summer.

Anexo II. Documentos anexados en el formato electrónico

En este anexo se incluye un listado con los documentos y programas que se encuentran anexados en el formato electrónico

Índice del Proyecto

1. INTRODUCCIÓN.....	1
1.1. JUSTIFICACION Y OBJETIVOS.....	2
1.2. ANTECEDENTES.....	3
2. DISEÑO DEL CIRCUITO DE ALIMENTACIÓN, INSTRUMENTACIÓN Y ACONDICIONAMIENTO.....	7
2.1. SENSORES.....	8
2.1.1. Galgas extensiométricas.....	8
2.2. CIRCUITO DE ACONDICIONAMIENTO.....	11
2.2.1. Puente de Wheatstone.....	11
2.2.2. Amplificador de instrumentación.....	12
2.2.3. Equilibrio y calibración de los sensores.....	13
2.2.4. Esquema del circuito de instrumentación y acondicionamiento.....	17
2.3. CIRCUITO DE ALIMENTACIÓN.....	18
2.3.1. Alimentación a +5V.....	19
2.3.2. Alimentación a 3.3V.....	19
2.3.3. Alimentación a -5V.....	19
2.3.4. Esquema del circuito completo de alimentación.....	20
3. PROCESADOR DE SEÑAL.....	22
3.1. CONFIGURACIÓN DEL OSCILADOR.....	24
3.2. CONFIGURACIÓN DEL CONVERTOR AD.....	26
3.3. CONFIGURACIÓN DE LA UART.....	29
3.4. CONFIGURACIÓN DE LOS PINES RECONFIGURABLES.....	31
3.5. CONFIGURACIÓN DE LOS PUERTOS I/O.....	31
3.6. ESQUEMA DEL CIRCUITO DEL PROCESADOR.....	32
4. COMUNICACIÓN CON EL PC.....	34
4.1. BLUETOOTH.....	35
4.2. MODULO PROMI-ESD02.....	36
4.2.1. Configuración del módulo.....	37
4.3. EMPAREJAMIENTO DE LOS DISPOSITIVOS BLUETOOTH.....	39
5. ELABORACIÓN Y DISEÑO DE LA PCB.....	45
5.1. PCB.....	46
5.2. DISEÑO DEL CIRCUITO IMPRESO.....	46
5.3. MÉTODOS PARA LA ELABORACIÓN DE CIRCUITOS IMPRESOS.....	49
6. PROGRAMAS REALIZADOS.....	50
6.1. PROGRAMA DSPIC.....	51

6.1.1.	Explicación de programa	51
6.1.2.	Diagrama de flujo del programa	52
6.1.3.	Programa dsPIC	54
6.2.	APLICACIÓN DE EJEMPLO	58
7.	PRESUPUESTO.....	62
8.	CONCLUSIONES.....	64
8.1.	LÍNEAS FUTURAS DE TRABAJO	65
9.	BIBLIOGRAFÍA.....	66
ANEXO I.....	69
A1.1.	CREACIÓN DE UN PROYECTO	70
A1.2.	DISEÑO DEL ESQUEMÁTICO	72
A1.3.	CREACIÓN DE LIBRERÍAS	78
A1.4	DISEÑO DE LA PCB.....	82
ANEXO II.....	84

1.Introducción

En este capítulo se hace una introducción en el ámbito de la posturografía y se establecen los objetivos del proyecto analizando los antecedentes y las necesidades.

1. Introducción

El Centre d'Estudis Tecnològics per a la Dependència i la Vida Autònoma (CETpD) es un centro de investigación aplicada y de transferencia de tecnología creado por la Universitat Politècnica de Catalunya (UPC) y la Fundació Hospital Comarcal Sant Antoni Abat (FHCSAA) con el propósito de responder a la demanda de investigación y desarrollo en el campo de la gerontecnología, los ambientes inteligentes, la robótica asistencial y la experiencia de usuario.

Las alteraciones de la marcha o el equilibrio y las caídas son las principales causas de discapacidad en la población anciana. Tales patologías afectan gravemente la calidad de vida de los ancianos; además, sus consecuencias tienen un alto impacto social y económico en nuestra sociedad. El área de Equilibrio, marcha y caídas del CETpD realiza estudios clínicos y epidemiológicos sobre este tipo de patologías con el fin de conocer sus causas y encontrar nuevas terapias que ayuden a paliarlas y/o combatirlas de forma efectiva.

En el laboratorio de movilidad MoMa, usando técnicas novedosas como acelerómetros o posturómetros, se realizan estudios relacionados con discapacidades funcionales y problemas de movilidad, y para ello, entre otros dispositivos se dispone de una plataforma de fuerza para el análisis del equilibrio y el paso (Balance Master), la cual permite realizar pruebas de equilibrio estático y análisis de la marcha.

1.1. JUSTIFICACION Y OBJETIVOS

El 98% de las pruebas realizadas con la plataforma de fuerza en el laboratorio MoMa, consisten en pruebas de equilibrio estático, y normalmente siempre las mismas, con lo que solo se aprovecha un 15% de las posibilidades de la plataforma.

El coste de la plataforma es superior a los 30.000€ con lo que la plataforma esta subutilizada y es difícil de amortizar. El tamaño que es superior a los 4'5 metros, condiciona el lugar donde ha de estar instalada que ha de ser un recinto suficientemente grande.

Por ello surge la idea de crear una plataforma de posturografía estática portátil con dimensiones, peso y precio reducidos, y este es el objeto de este proyecto. Para ello se utilizará una balanza comercial de bajo coste con 4 sensores, la cual se modificará añadiendo un microcontrolador en un circuito para adquirir la información de los sensores, analizar y tratar los datos, y transmitir los resultados a un ordenador. El proyecto seguirá la estructura que se muestra en la Figura 1

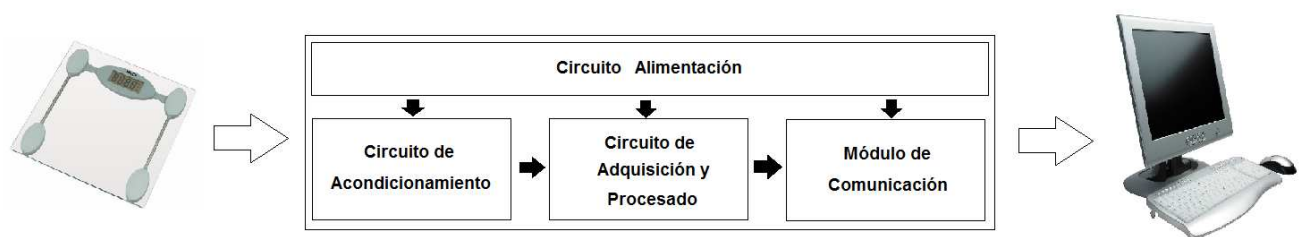


Figura 1 Esquema del proyecto

Los objetivos parciales a desarrollar en este proyecto de desarrollo de un sistema de posturografía portátil son:

- Diseño e implementación de un circuito de acondicionamiento, adquisición, tratamiento y transmisión de datos para una balanza comercial con 4 sensores
- Programación de la adquisición, tratamiento y transmisión de datos

- Programa de ejemplo

Al finalizar el proyecto se habrá conseguido un sistema de posturografía portátil de bajo costo y reducido tamaño que permitirá realizar las pruebas típicas de equilibrio estático en cualquier consulta.

1.2. Antecedentes

Este proyecto se basa en la elaboración de una plataforma de posturografía portátil. Antes de proceder con los aspectos técnicos de dicho proyecto, se hace necesaria una introducción para comprender en qué consiste la posturografía así como sus aplicaciones en los diferentes entornos, y un análisis de los equipos ya existentes.

La **posturografía** comprende todas las técnicas utilizadas para la evaluación del equilibrio y la postura mediante el cálculo de la posición del centro de gravedad del paciente en posición bípeda. Para ello se utilizan equipos de posturografía, que, dependiendo de sus condiciones, serán de posturografía dinámica o de posturografía estática.

La **posturografía estática** se basa en las técnicas utilizadas para la evaluación del equilibrio y la postura con el paciente en posición bípeda mediante una plataforma fija que calcula el desplazamiento del centro de gravedad del paciente.

En el caso de la **posturografía dinámica**, la plataforma es móvil (pudiendo inclinarse) y el entorno visual es también móvil.

Dichas plataformas se denominan Force Plate o Balance Board.

Se pueden encontrar equipos de posturografía en diferentes ámbitos como es el clínico, el deportivo o el lúdico.

En el ámbito clínico se utilizan estos equipos para el diagnóstico y rehabilitación de los pacientes con alteraciones del equilibrio y trastornos neurosensoriales. Para ello se realizan diferentes pruebas sobre la plataforma enmarcadas en el denominado Test de Romberg, que se basa en la premisa de que una persona necesita al menos dos de los tres siguientes sentidos para mantener el equilibrio estando de pie:

- La **propiocepción**: Informa al organismo de la posición de los músculos, es la capacidad de sentir la posición relativa de partes corporales contiguas.
- La **sensibilidad**: La capacidad de sentir el tacto, la presión o vibración.
- La **visión**: Es una de las principales capacidades sensoriales del ser humano, es la capacidad de interpretar el entorno y puede ser útil para monitorear los cambios en el equilibrio.

Para la realización de dicho test el sujeto debe colocarse en posición bípeda con los pies juntos sobre la plataforma y los brazos a lo largo del cuerpo; se le realizarán diferentes pruebas anulando o alterando alguno de los tres sentidos descritos anteriormente y se observará el desplazamiento del centro de gravedad del paciente. Las diferentes pruebas consisten en:

- Test de Romberg sobre superficie plana y ojos abiertos.
- Test de Romberg sobre superficie plana y ojos cerrados.
- Test de Romberg sobre gomaespuma y ojos abiertos.
- Test de Romberg sobre gomaespuma y ojos cerrados.

En el ámbito deportivo estos equipos se utilizan como entrenamiento y corrección de los movimientos en el Golf, como equipo de medida de la fuerza, altura y potencia en el salto en el fitness, etc.

Un ejemplo lúdico y actual sería el caso de la Wii Balance Board, accesorio de la videoconsola Nintendo Wii que detecta el peso y el balanceo del jugador y lo aplica en diferentes juegos como el esquí o el fitness.



Figura 2 Wii Balance Board.

A continuación se muestran distintos equipos de posturografía comerciales:

Balance Master, Neurocom®



Figura 3 Balance Master

Equipo de medida de la fuerza vertical ejercida en los pies del paciente (F_z), determina también las características de la marcha cuando el paciente camina sobre él, además de otras funciones como la habilidad de levantarse en el paciente cuando éste se encuentra sentado.

Dimensiones: 457.2x1524mm 18"x 60"

Portable multi component force plate 9286B, Kistler



Figura 4 Force Plate 9286B

Equipo de medida de la fuerza en los ejes X, Y y Z, para el análisis del paso y el equilibrio.

Dimensiones: 600x400x35 mm

Peso: 17,5 Kg

Rango de medida: F_X, F_Y : -2,5...2,5 kN

F_Z : 0...10 kN

Force Plate FP4060-07-100, Bertec



Figura 5 Force Plate FP4060-07-100

Equipo de medida de la fuerza en los ejes X, Y y Z, para el análisis del paso, del equilibrio, ergonómico, estático y dinámico.

Dimensiones: 600x400x75 mm

Peso: 38 Kg

Rango de medida: F_X, F_Y : 0...2,5 kN

F_Z : 0...5 kN

AccuGait, Amti



Figura 6 AccuGait

Equipo de medida portátil para el cálculo del equilibrio y el paso. Medida de la fuerza en los ejes X, Y y Z.

Dimensiones: 500x500x44 mm

Peso: 11,4 Kg

Rango de medida: F_X, F_Y : 0...445 N

F_Z : 0...2,67 kN

El estudio de los diferentes equipos de posturografía nos muestra que a pesar de la variedad de equipos de medida del equilibrio, pocos son los que, aún catalogándose como portátiles, cumplen con las medidas y peso que se ajustan a dicha característica.

2. Diseño del circuito de alimentación, instrumentación y acondicionamiento

En este capítulo se desarrollan los conceptos utilizados para el diseño de la parte analógica del hardware, formada por los sensores, el circuito de acondicionamiento y el circuito de alimentación

2. Diseño del circuito de alimentación, instrumentación y acondicionamiento

Como plataforma base, se ha utilizado una báscula de baño digital que proporciona 4 sensores analógicos (S1, S2, S3 y S4) situados tal y como se indica en la Figura 7.

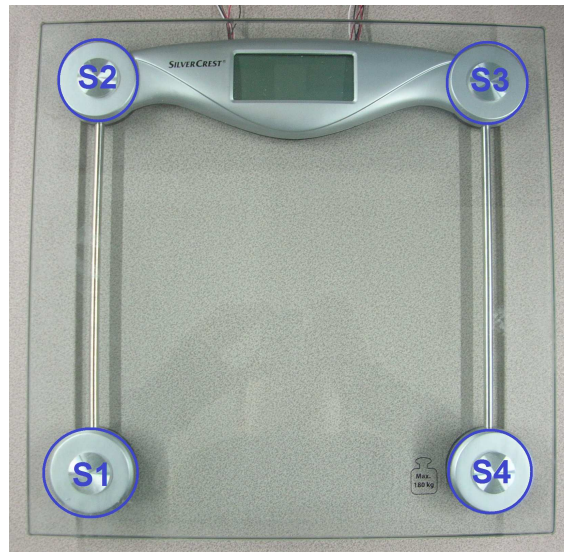


Figura 7 Situación de los sensores en la báscula

2.1. Sensores

Como se ha dicho, la báscula contiene 4 sensores. En la Figura 8 se muestra la apariencia de los sensores utilizados. Cada uno de estos sensores está formado por dos galgas extensiométricas activas con variaciones opuestas y montadas sobre soporte metálico.

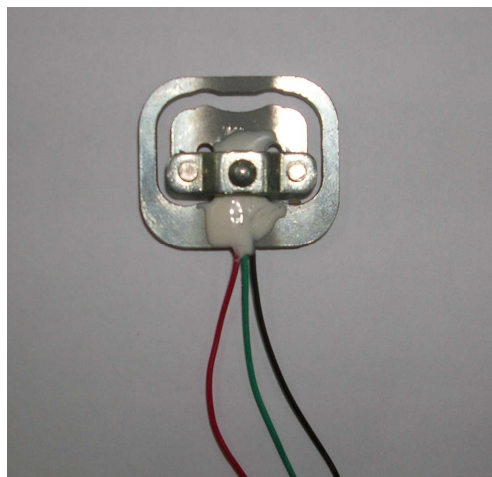


Figura 8 Apariencia del sensor

A continuación se desarrollan las características de las galgas extensiométricas.

2.1.1. Galgas extensiométricas

Las galgas extensiométricas (Figura 9) son sensores resistivos que se basan en la variación de la resistencia cuando son sometidos a un esfuerzo mecánico.

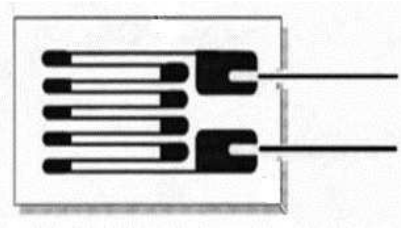


Figura 9 Galga extensiométrica

2.1.1.1. Principio de funcionamiento

Dado un hilo de metal homogéneo, de longitud l , sección S y resistividad ρ , y sujeto por los dos extremos. La resistencia asociada a este hilo de metal es:

$$R = \rho \cdot \frac{l}{S}$$

Si se define S como la sección cilíndrica del hilo, con diámetro d :

$$S = \frac{\pi \cdot d^2}{4}$$

Y sustituyendo esta expresión en la de la resistencia, tenemos:

$$R = \rho \cdot \frac{l}{\frac{\pi \cdot d^2}{4}}$$

Si se ejerce un esfuerzo longitudinal en los dos extremos del hilo, se experimentará un cambio en las magnitudes que intervienen en el valor de la resistencia, que producirán un cambio en el valor de esta:

$$\frac{\Delta R}{R} = \frac{\Delta \rho}{\rho} + \frac{\Delta l}{l} - 2 \frac{\Delta d}{d}$$

La relación que existe entre la variación de la longitud y de la sección del hilo, al someterlo a una fuerza de tracción o compresión viene determinada por el denominado coeficiente de Poisson, expresado en microdeformaciones (ppm):

$$\nu = - \frac{\Delta d / d}{\Delta l / l}$$

El factor que determina la sensibilidad de la resistencia a los cambios de longitud se denomina factor de galga:

$$K = \frac{\Delta R / R}{\Delta l / l} = 1 + 2\nu + \frac{\Delta \rho / \rho}{\Delta l / l}$$

Si se tiene en cuenta la fuerza F que se ejerce sobre el hilo y el módulo de Young E del material, la elongación por unidad de longitud es:

$$\varepsilon = \frac{F}{E \cdot S}$$

Por lo tanto el cambio en la resistencia será:

$$\Delta R = \frac{K \cdot R}{E \cdot S} \cdot F$$

Cuando se produce una deformación sobre una galga, el cambio en el valor de la resistencia puede ser debido al cambio en la longitud, la sección o la resistividad.

Así, dadas las ecuaciones anteriormente definidas, se puede afirmar que la sensibilidad de la galga aumenta con el factor de galga y el valor de la resistencia, y decrece con el módulo de Young y la sección.

Como lo que se pretende es medir las fuerzas en una zona concreta, el tamaño no puede ser muy grande, de manera que el hilo se dispone en zigzag y se sitúa sobre una película que transmita lo mejor posible las fuerzas al hilo. Asimismo, la galga debe estar aislada eléctricamente del objeto donde se mide y protegida del ambiente mediante un aislamiento protector tal y como se muestra en la Figura 10.

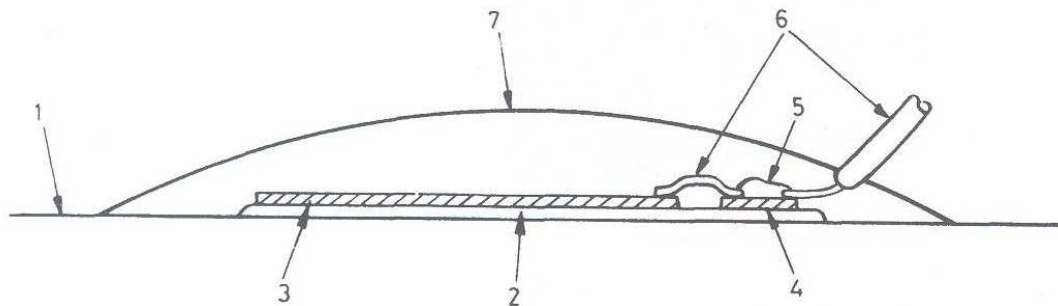


Figura 10 Montaje de la galga. 1 Sustrato donde se monta. 2 adhesivo. 3 Galga. 4 terminales (pads) para soldar. 5 soldadura. 6 hilos de conexión. 7 aislamiento protector.

Cabe destacar y tener en cuenta, que un factor que afecta a las galgas es la temperatura, afectando directamente a la resistividad y el módulo de elasticidad del material, a sus dimensiones y a las dimensiones del soporte. Por lo tanto, al producirse un cambio en la temperatura, aunque no se haya aplicado ningún esfuerzo sobre la galga, se producirá un cambio en el valor de la resistencia debido a la variación de las magnitudes que hemos citado.

Debido a que la temperatura es una fuente de interferencias, aparte de la influencia que la temperatura ambiente efectúa sobre las galgas, también hay que tener en cuenta el calentamiento de la galga provocado por la potencia que disipa cuando sobre ella circula una corriente eléctrica.

A pesar de estas limitaciones, debido a su pequeño tamaño, gran linealidad y baja impedancia, las galgas extensiométricas son uno de los sensores con mayor aplicación.

2.1.1.2. Tipos de galgas

Para la fabricación de galgas se pueden utilizar:

- Conductores metálicos, como son las aleaciones constantan, advance o isoelastic, las cuales tienen un bajo coeficiente de temperatura. Las galgas fabricadas con estos conductores tienen como ventajas su bajo coste, la gran variedad de topologías y pads de conexión de un tamaño suficiente que nos permite conectarlas con facilidad.
- Semiconductores como el silicio y el germanio. Las galgas fabricadas con estos materiales tienen un factor de galga más elevado que las metálicas, aún así, las galgas fabricadas con semiconductores son poco predominantes en el mercado debido a su elevado coste, mayor sensibilidad térmica y pads de conexión pequeños, que exigen una compleja soldadura mediante resinas conductoras o soldadura por ultrasonidos.

La Tabla 1 nos permite comparar las diferentes características de las galgas extensiométricas metálicas y las semiconductoras:

Parámetro	Metálicas	Semiconductoras
Margen de medida ($\mu\epsilon$)	0,1- 40000	0,001- 3000
Factor de galga	2	30 a 120
Resistencia (Ω)	120, 350, 600...5000	1000-5000
Tolerancia en la resistencia (%)	0,1 a 0,2	1-2
Tamaño (mm)	0,4-150 (estándar: 3-6)	1-5

Tabla 1 Características de las galgas metálicas i semiconductoras

2.2. Circuito de acondicionamiento

2.2.1. Puente de Wheatstone

Para obtener una señal eléctrica proporcional a la fuerza que se aplica sobre la galga se utiliza el puente de Wheatstone, el cual presenta la topología que se observa en la Figura 11. El método que se utiliza es el de medida por deflexión, que consiste en medir la diferencia de tensión entre las dos ramas (V_d) del puente alimentado a tensión o corriente constante. En nuestro caso el circuito se encontrará alimentado a una tensión constante de 5V.

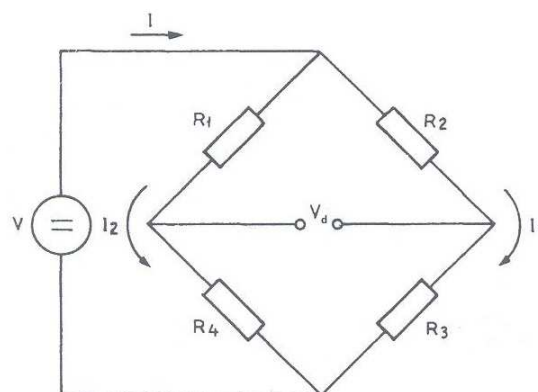


Figura 11 Topología Puente de Wheatstone

El cálculo de V_d del anterior circuito será:

$$V_d = \left(\frac{R_3}{R_2 + R_3} - \frac{R_4}{R_1 + R_4} \right) \cdot V$$

Para modelar el puente, partimos de que lo que nos interesa es que en condiciones iniciales V_d sea 0, por lo tanto $R_1/R_4 = R_2/R_3$. En nuestro caso, cada puente consta de dos galgas activas de $1K\Omega$ en R_2 y R_3 (configuración medio puente). Por lo tanto las resistencias a seleccionar para R_1 y R_4 deben tener igual valor para que se cumpla la relación. En nuestro caso las resistencias seleccionadas son de $1K\Omega$.

2.2.1.1. Configuración medio puente

La configuración medio puente consiste en un puente de Wheatstone con dos galgas activas. En nuestro caso el sensor está compuesto por las dos galgas activas, las cuales están colocadas en la misma rama del circuito, estas galgas experimentan deformaciones de igual amplitud pero de signo opuesto. Y en la otra rama que forma el puente de wheatstone se encuentran las dos resistencias de $1K\Omega$ tal y como se observa en la Figura 12.

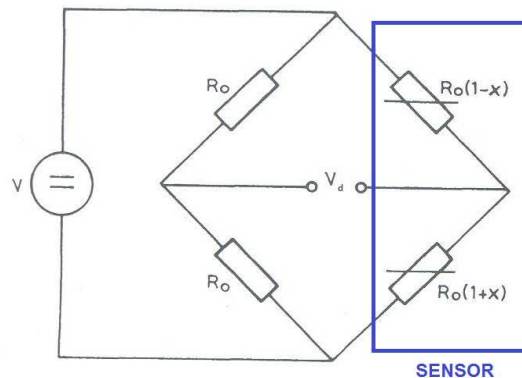


Figura 12 Configuración medio puente

Esta configuración nos permite doblar la sensibilidad del puente y obtener una tensión V_d lineal, de modo que:

$$V_d = V \cdot \frac{x}{2}$$

2.2.1.2. Montaje a tres hilos

Dado que en el circuito las galgas no se encuentran en el mismo lugar que el puente, la distancia de las galgas puede afectar en el cableado introduciendo resistencias parasitas que afecten en la medida. Es por ello que se utiliza el montaje a tres hilos, que consiste en utilizar cables de la misma longitud en la rama donde se encuentran las galgas de manera que los cambios que se producen en las resistencias del cableado se produzcan tanto en la parte inferior del puente como en la superior, de manera que se compensan los efectos de estas resistencias parasitas y no influyen en el valor de la tensión.

2.2.2. Amplificador de instrumentación

Un amplificador de instrumentación es un amplificador de tensión diferencial, de precisión, con una alta impedancia de entrada, alto rechazo en modo común (CMRR), impedancia de salida baja, tensiones y corrientes de offset bajas, y que sea capaz de trabajar con sensores que ofrecen señales eléctricas muy débiles.

Para la amplificación de las señales de los 4 sensores de la plataforma se ha seleccionado el amplificador de instrumentación AD620 que está basado en el modelo de amplificador de instrumentación con 3 amplificadores operacionales como el que se muestra a continuación:

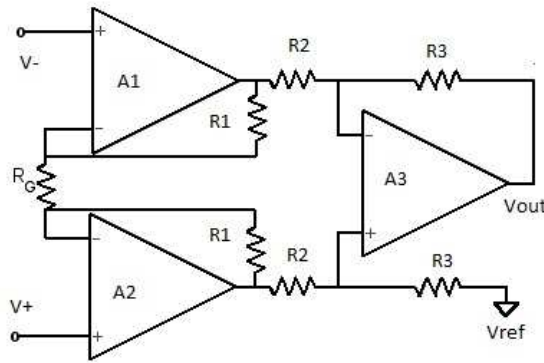


Figura 13 Amplificador de instrumentación

Las principales características que han servido como criterio de elección del AD620 son:

- Ganancia ajustable con resistencia externa.
- Rangos de ganancia de 1 a 1000.
- Bajo consumo, 1.3mA max., perfecto para aplicaciones portátiles.
- Bajo ruido.
- Bajo coste.

Para el ajuste de la ganancia, el fabricante proporciona las siguientes ecuaciones:

$$G = \frac{49.4K\Omega}{R_G} + 1$$

Por tanto, el valor de R_G en función de la ganancia será:

$$R_G = \frac{49.4K\Omega}{G - 1}$$

Esta ecuación se utilizará para la calibración de los 4 sensores.

2.2.3. Equilibrio y calibración de los sensores

2.2.3.1. Equilibrio

Idealmente, en condiciones iniciales, y sin que ejercer ninguna fuerza sobre las galgas, la tensión de salida V_d debería ser 0. Esto no se produce en la realidad debido a las tolerancias y el comportamiento térmico de las diferentes resistencias, por eso será necesario un ajuste inicial del cero mediante un potenciómetro situado tal y como indica la Figura 14.

Además de este ajuste manual del cero, éste también se corregirá por programa.

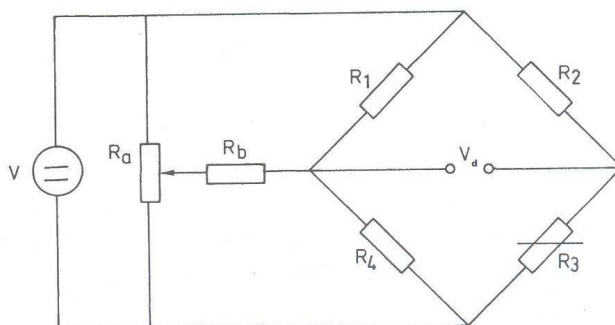


Figura 14 Circuito de ajuste de cero con potenciómetro

2.2.3.2. Calibración

La calibración consiste en establecer la correspondencia entre los valores del sensor y los valores de la magnitud a medir. Para ello se elaborará una recta de calibración para cada uno de los cuatro sensores de los que dispone la plataforma.

Aunque el método shunt sea el más habitual para evaluar la salida del sensor mediante la adición de resistencias en paralelo con las galgas activas que simulen una variación en el valor de la resistencia y por lo tanto una deformación, en nuestro caso esta técnica no es útil ya que desconocemos el factor de la galga.

Por tanto, en nuestro caso, para obtener la relación de los valores de tensión de la salida de los sensores con los kilogramos aplicados, se colocaran pesos de 1Kg, 2Kg y 3 Kg sobre cada uno de los sensores de la báscula individualmente. Los valores obtenidos, restando el offset que produce el cristal de la plataforma, son los siguientes:

Peso (Kg)	Vd (mV) sensor 1	Vd (mV) Sensor 2	Vd (mV) Sensor 3	Vd (mV) Sensor 4
0	0,063	0,088	0,081	0,045
1	0,156	0,17	0,178	0,122
2	0,249	0,255	0,273	0,213
3	0,063	0,088	0,081	0,045

Tabla 2 Tensiones diferenciales Vd de los sensores en función del peso

A continuación se muestra la recta de regresión de cada sensor, la ecuación de la recta y el índice de correlación, que como se observa es muy cercano a 1 en todos los casos:

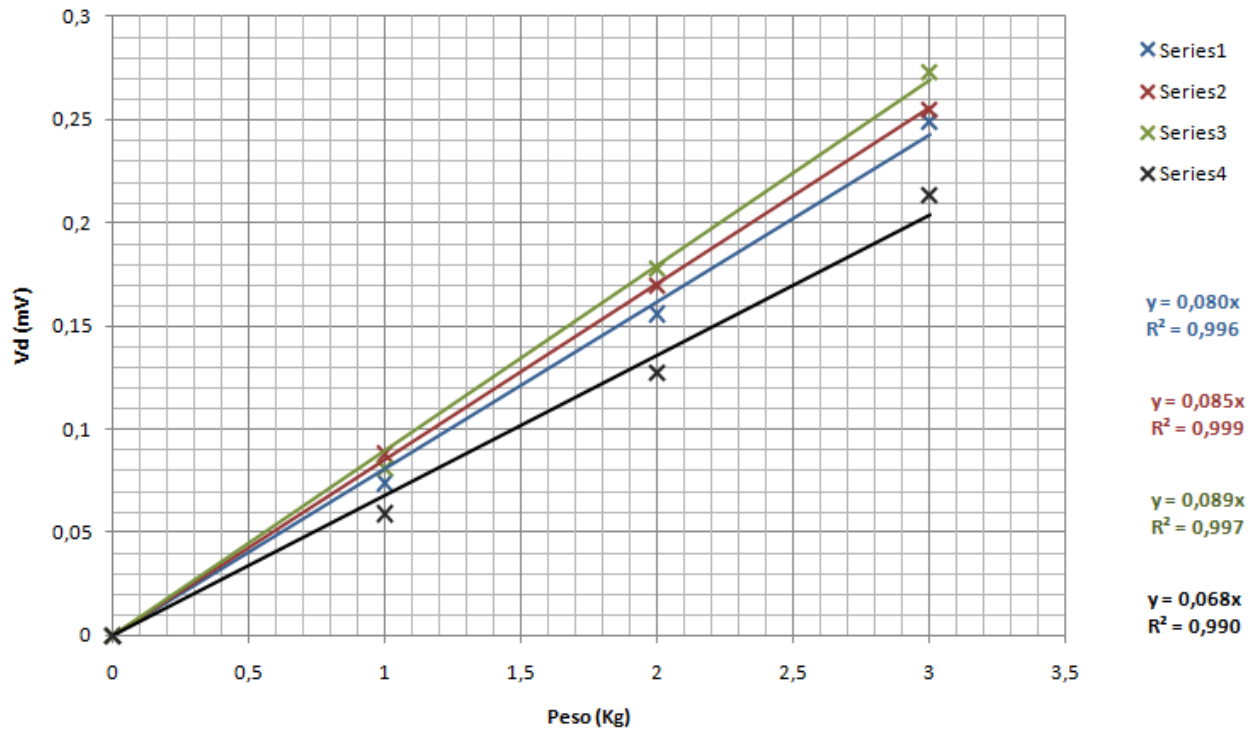


Figura 15 Grafica de relación entre el Peso y la tensión diferencial V_d

El último paso de la calibración es el ajuste de la ganancia, como se ha comentado anteriormente, para ello se ha utilizado un amplificador de instrumentación AD620 que nos permite ajustar la ganancia (G) mediante el ajuste de la resistencia R_G (potenciómetro):

$$R_G = \frac{49.4K\Omega}{G - 1}$$

Para el cálculo de G :

$$V_{O_{MAX}} = G \cdot V_{d_{max}} - V_{offset}$$

$$G = \frac{V_{O_{MAX}} - V_{offset}}{V_{d_{max}}}$$

V_{offset} será 0 ya que como anteriormente ya hemos realizado ajuste de cero no hay offset

Establecemos que el rango de medida de cada sensor será de 0Kg/50Kg, por lo tanto el peso máximo se corresponderá con el valor de tensión de salida máximo $V_{O_{max}}$. Para este proyecto nos interesa que las tensiones de salida máximas en los amplificadores sean 3.3V, ya que la tensión que soportan los pines del dsPIC al que van conectadas es de entre -0.3V a 3.6V.

Para el cálculo de $V_{d_{max}}$ se utilizarán las ecuaciones de los cuatro sensores para el caso de $x=50Kg$:

Sensor 1:

$$y = 0.080x = 0.080 \times 50 = 4 \rightarrow V_{d_{max}} = 4mV$$

Sensor 2:

$$y = 0.085x = 0.085 \times 50 = 4.25 \rightarrow V_{d_{max}} = 4.25mV$$

Sensor 3:

$$y = 0.089x = 0.089 \times 50 = 4.45 \rightarrow V_{d_{\max}} = 4.45mV$$

Sensor 4:

$$y = 0.068x = 0.068 \times 50 = 3.4 \rightarrow V_{d_{\max}} = 3.4mV$$

Ahora ya se puede calcular la ganancia de los 4 sensores:

Sensor 1:

$$G = \frac{3.3V}{4mV} = 825 \rightarrow R_G = \frac{49.4K\Omega}{825 - 1} = 59.95\Omega$$

Sensor 2:

$$G = \frac{3.3V}{4.25mV} = 776.47 \rightarrow R_G = \frac{49.4K\Omega}{776.47 - 1} = 63.70\Omega$$

Sensor 3:

$$G = \frac{3.3V}{4.5mV} = 741.57 \rightarrow R_G = \frac{49.4K\Omega}{741.57 - 1} = 66.71\Omega$$

Sensor 4:

$$G = \frac{3.3V}{3.4mV} = 970.59 \rightarrow R_G = \frac{49.4K\Omega}{970.59 - 1} = 50.95\Omega$$

Estos son los valores que hay que ajustar en los potenciómetros conectados a los pines R_G de cada amplificador.

A continuación se muestra la grafica que relaciona la tensión de entrada V_d con la de salida V_o del amplificador, para cada sensor:

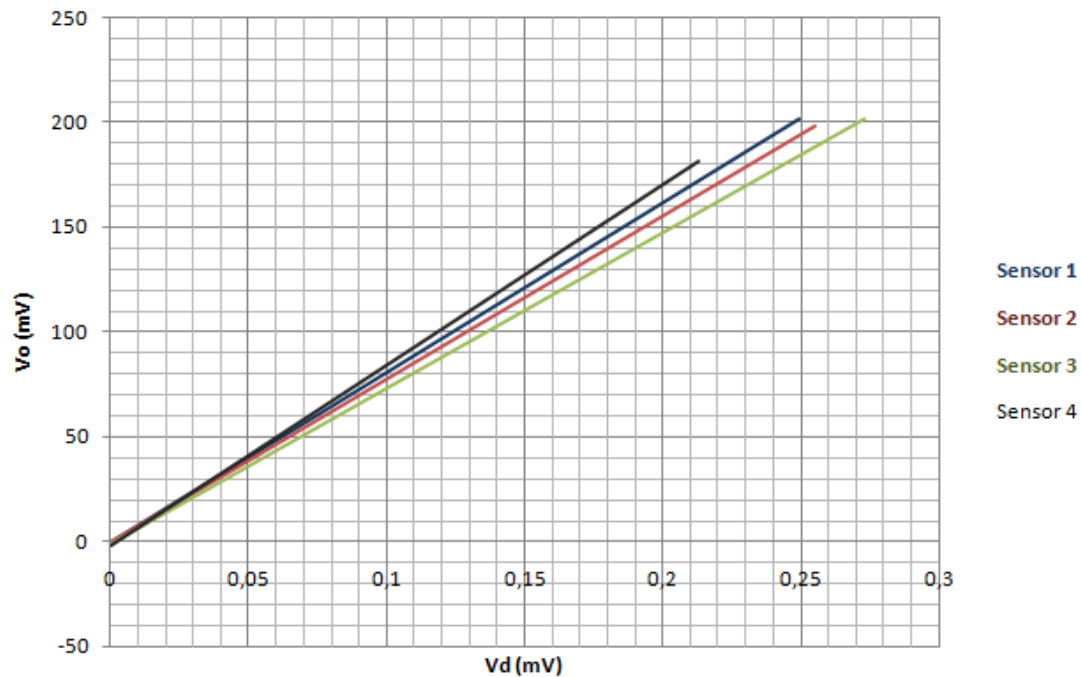


Figura 16 Gráfica V_d - V_o

Como se observa, los valores de las diferentes rectas se aproximan entre ellos con pendientes muy cercanas, objetivo que se pretendía en la calibración.

2.2.4. Esquema del circuito de instrumentación y acondicionamiento

Como se observa, para el primer modulo, R1 y R2 ($1K\Omega$) completan el puente de wheatstone junto con el medio puente de galgas conectado en la regleta Y1. El puente está alimentado a 5V y el potenciómetro Pot10 se utiliza para el ajuste del cero. El amplificador de instrumentación AI1 está alimentado a $\pm 5V$. La entrada positiva del amplificador AI1 corresponde al punto medio de la rama formada por las dos galgas, y que se corresponde con el pin 2 de la regleta Y1, la entrada negativa del AI1 corresponde al punto medio entre las dos resistencias R1 y R2. El potenciómetro Pot1 se utilizara para ajustar la ganancia del AI1 con el valor de R_G anteriormente calculado. La salida del amplificador AI1 ira conectada a la entrada analógica AN4 del dsPIC.

Y, lo mismo ocurrirá con el resto de módulos tal y como se puede observar en el esquema.

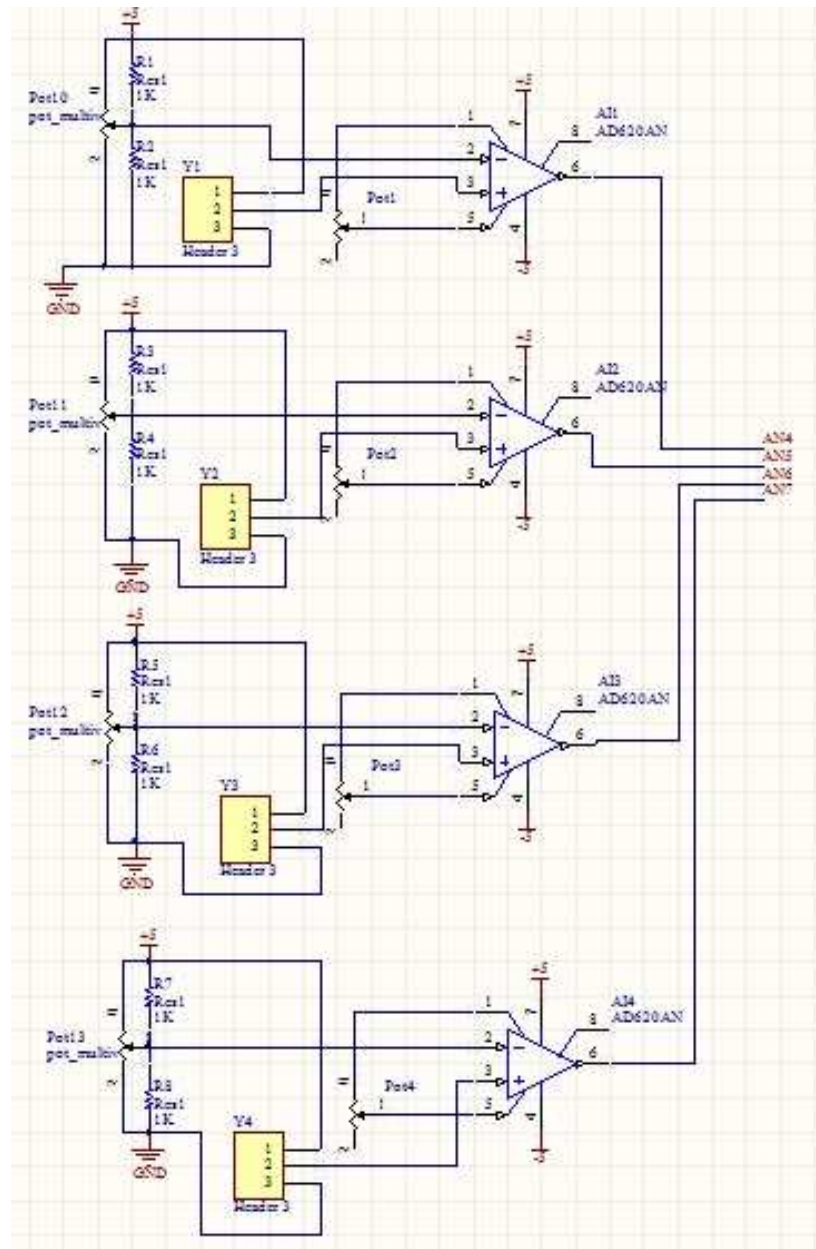


Figura 17 Esquema del circuito de instrumentación y acondicionamiento

2.3. Circuito de alimentación

El diseño del circuito de alimentación es un paso fundamental para la autonomía del dispositivo ya que como característica principal, nos interesa que el dispositivo sea portátil, y por lo tanto ligero. Para reducir espacio y peso utilizaremos una pila alcalina de 9V.

Una vez definido el tipo de alimentación deberemos establecer la tensión de alimentación que corresponda a cada parte del circuito:

- Instrumentación y acondicionamiento: Se alimentaran los puentes de Wheatstone a 5V, y los amplificadores de instrumentación a $\pm 5V$ ya que necesitan alimentación simétrica.
- Adquisición de datos y transmisión: Esta parte del circuito, que comprende el dsPIC y el modulo bluetooth, necesita una tensión de alimentación de 3.3V debido a las características eléctricas de los componentes.

A continuación se detalla el diseño de los circuitos de alimentación y las características para obtener los voltajes de +5V, -5V y +3.3V a partir de una tensión de alimentación de +9V.

2.3.1. Alimentación a +5V

Para obtener una tensión de alimentación estable de 5V a partir de una pila de 9V, se utilizará un regulador de tensión LM7805, el cual nos proporcionará una salida de 5V para una entrada de entre 7V hasta 20V.

Para obtener una salida fija regulada de 5V, el fabricante del dispositivo recomienda la siguiente disposición:

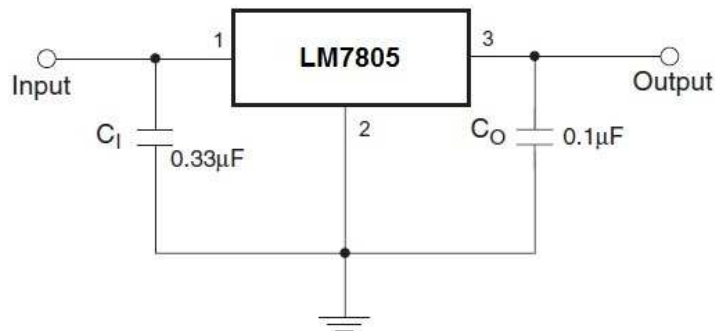


Figura 18 Regulador con salida fija a 5V

2.3.2. Alimentación a 3.3V

Para obtener una tensión de alimentación estable a 3.3V utilizaremos el regulador TC1262-33VAB, el cual proporciona una salida regulada a $3.3V \pm 0.5\%$ y una corriente fija de 500mA.

Para ello el dispositivo se dispondrá de la siguiente forma:

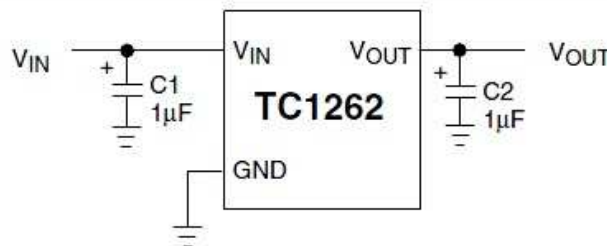


Figura 19 Diseño circuito regulador 3.3V

V_{IN} ira conectado a la salida del regulador LM7805, +5V, ya que el dispositivo soporta como máximo una tensión de entrada $V_{IN} = 6.5V$.

2.3.3. Alimentación a -5V

Para obtener una tensión negativa a partir de una pila de +9V necesitaremos algún dispositivo que nos invierta la tensión, para ello utilizo el TC7660, un convertidor de tensiones desde +1.5V hasta +10V a los correspondientes -1.5V hasta -10V, eliminando el uso de inductores, con su coste, tamaño e interferencias electromagnéticas asociadas.

Para que el dispositivo funcione como un simple convertidor a tensión negativa, el circuito debe ser:

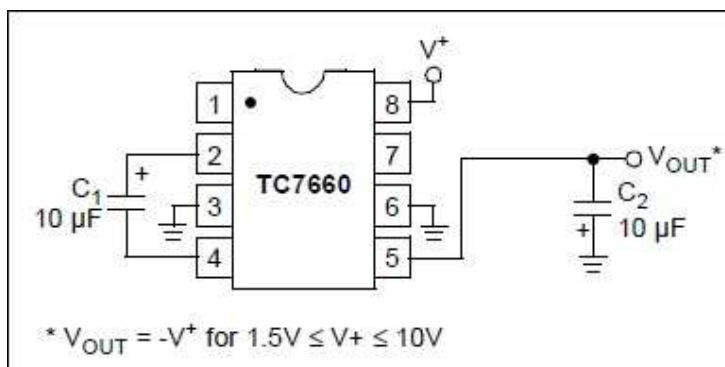


Figura 20 Diseño para el convertidor de tensión

En la salida de tensión negativa V_{OUT} se incorporará un regulador a -5V, ya que en el circuito $V^+ = +9V$, y por tanto $V_{OUT} = -9V$.

El regulador que se utilizará es el L7905CV, que proporciona una tensión de -5V para tensiones de entrada de entre -7V a -20V. Para ello el dispositivo se dispondrá de la siguiente forma:

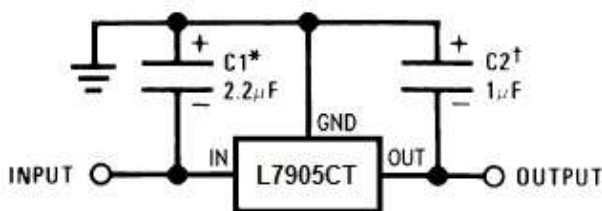


Figura 21 Regulador negativo con salida fija a -5V

2.3.4. Esquema del circuito completo de alimentación

En la Figura 22 se muestra el circuito completo utilizado en el proyecto para obtener las tensiones +5V, -5V y +3.3V.

En la regleta Y0 irá conectada la pila de 9V. Se utilizará el interruptor S1 para encender y apagar el circuito.

El LED L0 nos indicará si hay alimentación principal (9V) y la resistencia R0 se utiliza como limitador de tensión para proteger al LED.

Directamente después del interruptor se encuentra el regulador a 5V Reg1 por cuya entrada IN entran los 9V obteniendo en la salida OUT los 5V. También se encuentra el convertidor a tensiones negativas TC7660 por cuya entrada V^+ entrarán los 9V y obteniendo los -9V en la salida V_o , a la cual irá conectada la entrada IN del regulador negativo a 5V Reg 2., y que proporcionará en su salida OUT una tensión de -5V.

La salida OUT de regulador a 5V Reg1, además de proporcionar la alimentación de 5V al circuito, irá conectada a la entrada IN del regulador a 3.3V Reg3, que proporcionará en su salida OUT una tensión de 3.3V.

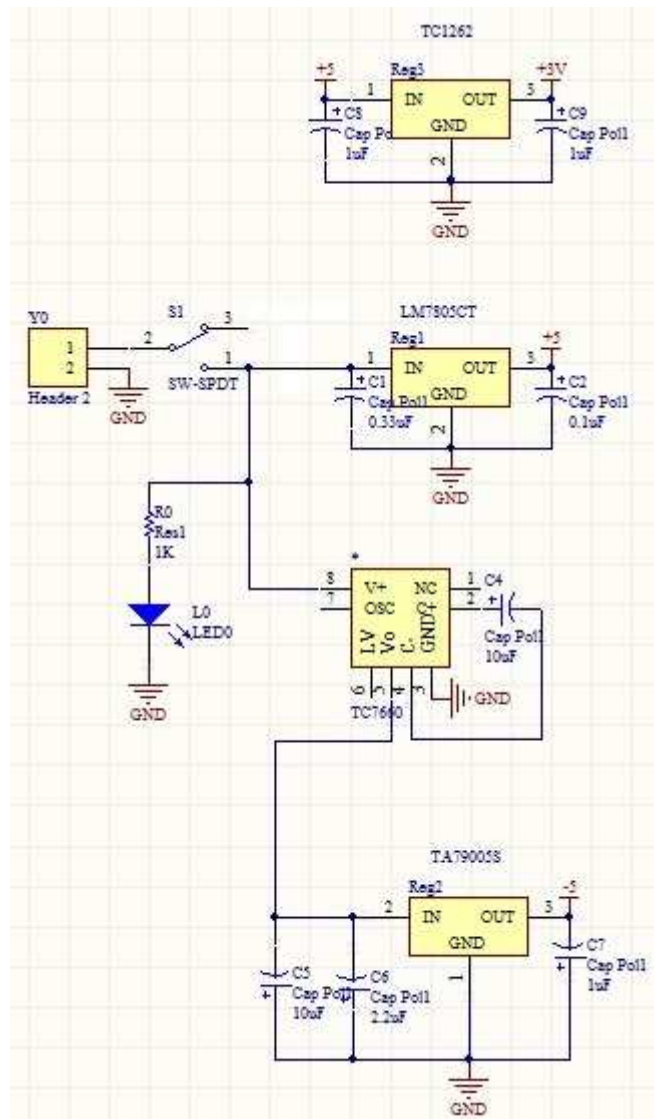


Figura 22 Esquema del circuito de alimentación

3. Procesador de señal

En este capítulo hace referencia a la parte digital del circuito, y en él se definen las características del microcontrolador y se desarrollan los conceptos utilizados para su configuración según los requisitos deseados para el proyecto

3. Procesador de señal

Para el procesamiento de las señales se utilizará el dsPIC33FJ32GP204 con 44 pines (Figura 23), que es un avanzado microcontrolador de 16 bits del fabricante Microchip entre cuyas características destacan:

- Gran velocidad, hasta 40 Millones de Instrucciones por Segundo (MIPS).
- Bajo consumo, alimentación a 3.3V.
- Opciones del clock: externo, RC interno, cristal, etc.
- 16 registros de propósito general de 16 bits

Interrupciones:

- Hasta 21 fuentes de interrupción
- Hasta 3 interrupciones externas
- 7 niveles de prioridad programables
- 35 pines I/O digitales programables

Periféricos:

- Módulos de comunicaciones: I2C, SPI, UART.
- Conversor analógico-digital (ADC)
- Comparador de 16 bits
- Contador de 16 bits
- Hasta 3 timers de 16 bits

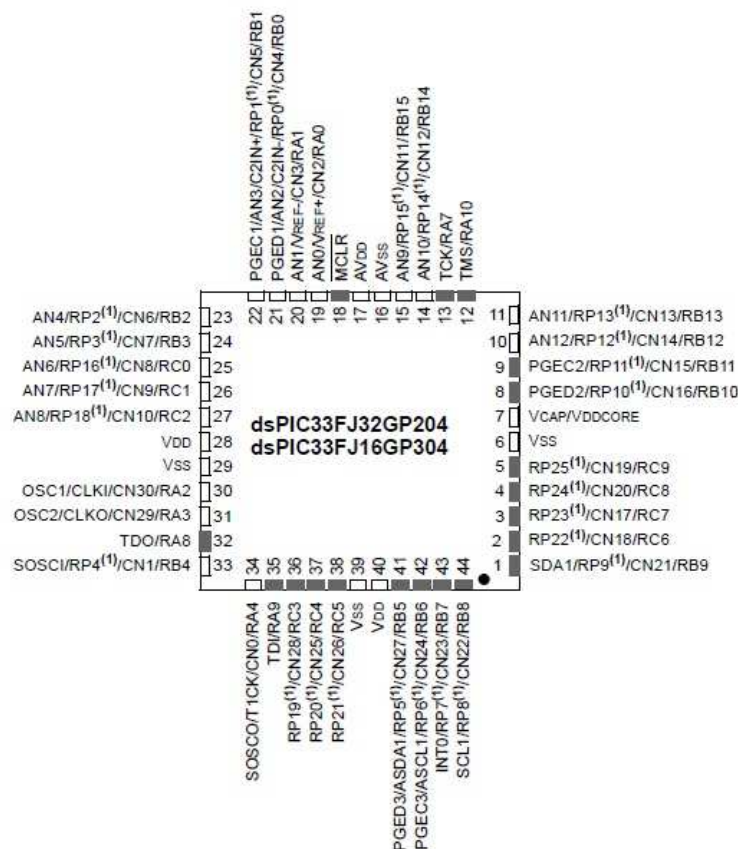


Figura 23 Diagrama de pines del dsPIC33FJ32GP204

Para el procesado y tratado, los datos obtenidos de los 4 AD620 van a parar a 4 entradas analógicas (AN4, AN5, AN6 y AN7) de el dsPIC33FJ32GP204 (Figura 23), el cual mediante su modulo conversor analógico-digital (ADC), convertirá las señales de los 4 sensores y las almacenará hasta obtener 10 valores de cada sensor, entonces procederá a hacer la media de los 10 valores de cada sensor, y se creará una tabla con los valores en ASCII de cada sensor precedidos por el correspondiente identificador (S1, S2, S3 o S4) y seguidos por un final de línea (LF) y retorno del carro (CR). Y a continuación se procederá a la transmisión de toda la tabla. Para ello se han utilizado los módulos ADC y UART y se han hecho uso de interrupciones para indicar conversión completada y transmisión realizada. La configuración de los módulos así como la del oscilador se definen a continuación.

3.1. Configuración del oscilador

El sistema oscilador del dsPIC33FJ32GP204 dispone de 4 Opciones de oscilador interno y externo, y Phase-Locked Loop (PLL) para aumentar la frecuencia de operación. En nuestro caso nos interesa aprovechar el oscilador interno FRC que proporciona 7.37MHz sin necesidad de cristal externo, y el PLL para conseguir una frecuencia de operación de 40MIPS. La fórmula para configurar el PLL es:

$$F_{OSC} = F_{IN} \times \left(\frac{M}{N1 \times N2} \right)$$

Donde:

$$F_{IN} = 7,37MHz$$

$$F_{OSC} = F_{CY} \times 2 = 40000000 \times 2 = 80MHz$$

Por tanto:

$$\frac{80000000}{7370000} = 10.855 = \left(\frac{M}{N1 \times N2} \right)$$

Tendremos entonces que ajustar los valores de M, N1 y N2 para obtener el valor más cercano a 10.855:

M=43

N1=2

N2=2

$$\frac{43}{2 \times 2} = 10.75$$

A continuación se detallan los registros para configurar según los requisitos:

FOSCSEL:

---	---	---	---	---	---	---	---
0	0	0	0	0	0	0	0
bit 15				bit 8			
IESO	---	---	---	---	FNOSC		
0	0	0	0	0	0	0	1

bit 7	bit 0
-------	-------

FNOSC: Selección del oscilador FRC interno con PLL (001)(FRCPLL).

Para la configuración de este registro podemos utilizar la siguiente consigna:

_FOSCSEL(FNOSC_FRCPLL)

FOSC:

---	---	---	---	---	---	---	---
0	0	0	0	0	0	0	0
bit 15						bit 8	
FCKSM		IOL1WAY	---	---	OSCIOFNC	POSCMD	
0	1	0	0	0	0	0	0
bit 7						bit 0	

En este registro nos interesan los bits:

FCKSM: Clock switching activado y Fail-Safe Clock Monitor desactivado (01)(CSECMD).

OSCIOFNC: OSC2 son pines I/O de propósito general (0)(OFF).

Para la configuración de este registro podemos utilizar la siguiente consigna:

_FOSC(FCKSM_CSEMD & OSCIOFNC_OFF)

PLLFBD:

---	---	---	---	---	---	---	PLLDIV
0	0	0	0	0	0	0	0
bit 15							bit 8
PLLDIV							
0	0	1	0	1	0	0	1
bit 7							bit 0

PLLDIV: Corresponde al valor M calculado anteriormente menos 2: $43-2=41$.

CLKDIV:

ROI	DOZE			DOZEN	FRCDIV		
0	0	0	0	0	0	0	0

bit 15								bit 8	
PLLPOST		---	PLLPRE						
0	0	0	0	0	0	0	0		
bit 7								bit 0	

En este registro nos interesan los bits:

PLLPOST: Corresponde al valor N2 calculado anteriormente menos 2: $2-2=0$.

PLLPRE: Corresponde al valor N1 calculado anteriormente menos 2: $2-2=0$.

OSCCON:

---	COSC			---	NOSC		
0	0	0	0	0	0	0	1
bit 15				bit 8			
CLKLOCK	IOLOCK	LOCK	---	CF	---	LPOSCEN	OSWEN
0	0	0	0	0	0	0	1
bit 7				bit 0			

En este registro nos interesan los bits:

COSC: Bits de lectura, nos interesa para saber si el reloj seleccionado es el oscilador FRC interno con PLL(001).

NOSC: Selección del oscilador FRC interno con PLL (001).

LOCK: Bit de lectura, nos interesa para saber si se ha fijado el PLL (1).

OSWEN: Activar (1) la selección especificada en NOSC.

3.2. Configuración del conversor AD

El dsPIC 33FJ32GP204 dispone de más de 13 entradas al modulo ADC. El modulo puede ser configurado como ADC de 10 bits i 4 canales o ADC de 12 bits y un solo canal. Aunque la configuración de 4 canales sería la mejor opción ya que permitiría realizar un muestreo simultaneo de las 4 entradas analógicas, con la configuración de 12 bits y un solo canal se gana mayor resolución, y aunque se tenga que realizar un escaneo de las 4 entradas la variación de tiempo entre el muestreo de la primera entrada hasta la cuarta es muy negligible.

El modo 12bits-1canal tiene las siguientes características:

- Pines de entrada para voltajes de referencia externos
- Velocidades de conversión de más de 500Ksps
- Modo de escaneo automático del canal
- Conversión por aproximación sucesiva (SAR)
- Modo de llenado del buffer seleccionable

- 4 Opciones de formato de los datos de salida: Signed Fractional, Fractional, Signed integer, Integer
- El resultado de la conversión se almacena en buffers de 16 bits
- Funcionamiento en modo CPU Sleep y Idle

Para el programa me interesa configurar el ADC con las siguientes características:

- Modo 12bits-1canal.
- Escaneo de 4 entradas analógicas: AN4, AN5, AN6, AN7
- Formato de los datos de salida: Integer
- Auto-conversión
- Muestreo Automático
- Tensión de referencia en la conversión: AV_{DD} , AV_{SS}
- Interrupción cada 4 secuencias de muestreo-conversión
- Reloj interno para la conversión
- Tiempo de muestreo automático: 16TAD

A continuación se detallan los registros utilizados para la configuración del ADC:

AD1CON1: 0x04E4

ADON	---	ADSIDL	---	---	AD12B	FORM	
0	0	0	0	0	1	0	0
bit 15						bit 8	
SSRC			---	SIMSAM	ASAM	SAMP	DONE
1	1	1	0	0	1	0	0
bit 7						bit 0	

En este registro nos interesan los siguientes bits:

ADON: Bit para activación (1) o desactivación (0) del modulo ADC. Para la configuración del modulo debe estar desactivado.

AD12B: Bit de selección del modo de trabajo (1): 12bits-1canal.

FORM: Formato de los datos de salida (00): Integer.

SSRC: Contador interno finaliza el muestreo y empieza la conversión (auto-conversión) (111).

ASAM: El muestreo empieza inmediatamente después de la última conversión (1).

SAMP: Bit para la activación del muestreo, como ASAM=1, este bit se activa automáticamente por hardware.

DONE: Bit de lectura sobre el estado de la conversión. Si DONE=0 la conversión no ha empezado o está en progreso, si DONE=1 el ciclo de conversión ha acabado.

AD1CON2: 0x040C

VCFG			---	---	CSCNA	CHPS	
0	0	0	0	0	1	0	0
bit 15						bit 8	
BUFS	---	SMPI				BUFM	ALTS
0	0	0	0	1	1	0	0
bit 7						bit 0	

En este registro nos interesan los siguientes bits:

VCFG: Tensión de referencia en la conversión: A_{VDD} , A_{VSS} (000).

CSCNA: Escaneo de las entradas usando el canal 0 (1).

BUFS: Bit de lectura sobre el estado del llenado del buffer. Si BUFS=0 el ADC está llenando la primera mitad del buffer, si BUFS=1 el ADC está llenando la segunda mitad del buffer.

SMPI: Numero secuencias de muestreo-conversión por interrupción: Se produce una interrupción después de cuatro secuencias de muestreo-conversión (0011).

BUFM: Bit para la selección del modo de llenado del buffer: Siempre empieza a llenar el buffer en la dirección de inicio (0).

ALTS: Channel Scan Logic selecciona la entrada analógica a convertir.

AD1CON3: 0x8E00

ADRC	---	---	SAMC				
1	0	0	0	1	1	1	0
bit 15			bit 8				
ADCS							
0	0	0	0	0	0	0	0
bit 7			bit 0				

En este registro nos interesan los siguientes bits:

ADRC: Bit para la selección del reloj para la conversión: Reloj RC interno (1).

SAMC: Tiempo de muestreo automático: 16TAD (01110).

AD1CSSL: 0x00F0

---	---	---	CSS12	CSS11	CSS10	CSS9	CSS8
-----	-----	-----	-------	-------	-------	------	------

0	0	0	0	0	0	0	0
bit 15							bit 8
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
1	1	1	1	0	0	0	0
bit 7							bit 0

CSSx: Seleccionamos las entradas Anx como entradas a escanear (An7, An6, An5, An4).

AD1PCFGL:

---	---	---	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
0	0	0	0				
bit 7							bit 0

PCFGx: Con PCFGx=0 configuramos el pin x como entrada analógica.

3.3. Configuración de la UART

El dsPIC 33FJ32GP204 dispone de un modulo UART, que es un canal de comunicación asíncrono y full-duplex que nos permite comunicarnos con periféricos y ordenadores usando protocolos como el RS-232, el cual nos interesa para transmitir y recibir mediante el modulo Bluetooth. Del modulo UART nos interesan las siguientes características:

- Modo de transmisión y recepción de datos de 8bits a través de los pines UxTX y UxRX. Full-duplex.
- Paridad par, impar o sin paridad.
- Un bit de stop
- Generador de Baud Rate integrado con Prescaler de 16bits.
- 4 Buffers FIFO de transmisión de datos.
- 4 Buffers FIFO de recepción de datos.
- Interrupciones de transmisión y recepción.

Para el programa me interesa configurar la UART con las siguientes características:

- Baud Rate=9600
- Modo 8 bits, 1 bit de stop y sin paridad
- Utilizar únicamente los pines: U1TX y U1RX

- Interrupción cuando el buffer de transmisión esté vacío (se ha transmitido un carácter al registro transmit shift)

A continuación se detallan los registros para la configuración de la UART1:

U1MODE: 0x0000

UARTEN	---	USIDL	IREN	RTSMD	---	UEN	
0	0	0	0	0	0	0	0
bit 15						bit 8	
WAKE	LPBACK	ABAUD	URXINV	BRGH	PSDEL		STSEL
0	0	0	0	0	0	0	0
bit 7						bit 0	

Para nuestra configuración únicamente nos interesan los siguientes bits:

UARTEN: Bit para la activación y desactivación de la UART1. Durante la configuración de ésta debe estar desactivada (0).

UEN: Se activan únicamente los pines: U1TX y U1RX (00).

PSDEL: Selección para datos de 8 bits sin paridad (00).

STSEL: Selección para un bit de stop (0).

U1STA: 0x8400

UTXISEL1	UTXINV	UTXISEL0	---	UTXBRK	UTXEN	UTXBF	TRMT
1	0	0	0	0	0	0	0
bit 15						bit 8	
URXISEL		ADDEN	RIDLE	PERR	FERR	OERR	URXDA
0	0	0	0	0	0	0	0
bit 7						bit 0	

Para nuestra configuración únicamente nos interesan los siguientes bits:

UTXISEL: Se generará una interrupción cuando se transmita un carácter al registro Transmit Shift y por lo tanto el buffer de transmisión está vacío (UTXISEL1=1 UTXISEL0=0).

UTXEN: Transmisión activada en la UART1, el pin U1TX está controlado por la UART1. Este pin se debe activar después de haber activado la UART (UARTEN)

U1BRG: 0x0103

BRG

0	0	0	0	0	0	0	1
bit 15							bit 8
BRG							
0	0	0	0	0	0	1	1
bit 7							bit 0

BRG: Divisor del Baud Rate, se calcula a partir de las siguientes formulas:

$$U1BRG = \frac{F_{CY}}{16 \times BaudRate} - 1$$

Como hemos visto en la configuración del oscilador, $F_{CY}=40\text{MHz}$, y queremos un $BaudRate=9600$:

$$U1BRG = \frac{40000000}{16 \times 9600} - 1 = 259 = 0x0103$$

3.4. Configuración de los pines reconfigurables

Como hemos visto en la configuración de la UART1 se utilizan los pines U1TX y U1RX para la transmisión y recepción de datos, estos pines son remapables o reconfigurables, lo que significa que, mediante la configuración de unos registros podemos ubicarlos en diferentes pines.

- **Configuración del pin 43 como U1TX:**

Como vemos en la diagrama de pines del dsPIC (Figura 23) el pin 43 se corresponde con RP7.

Y, basándonos en lo que establece el datasheet del dsPIC, sabemos que introduciendo el valor 00011 en RP7R (para RP7), el cual se encuentra en el registro RPOR3, configuraremos el pin 43 como U1TX.

Para hacerlo:

RPOR3bits.RP7R=0b00011;

- **Configuración del pin 44 como U1RX:**

Como podemos ver en el diagrama de pines de la Figura 23 el pin 44 se corresponde con RP8.

Y, basándonos en lo que establece el datasheet del dsPIC, sabemos que introduciendo el valor 8 (correspondiente a RP8) en los bits U1RXR del registro RPINR18, configuraremos el pin 44 como U1RX. Para hacerlo:

RPINR18bits.U1RXR= 0b01000;

3.5. Configuración de los puertos I/O

En este apartado se configuran como puertos de entrada las 4 Entradas analógicas AN4, AN5, AN6, AN7, que se corresponden con RB2, RB3, RC0 y RC1 (pines 23, 24, 25 y 26) como observamos en el diagrama de pines de la Figura 23.

Los registros TRISA, TISB, TRISC permiten configurar como puertos de entrada o salida.

Introduciendo un 1 en el correspondiente TRISx el pin Rx se configurará como entrada, e introduciendo un 0 se configurará como salida.

Los pines RB2 y RB3 corresponden a los bits 2 y 3 del registro TRISB. Para configurarlos como entradas: TRISB=0x000C

Los pines RC0 y RC1 corresponden a los bits 0 y 1 del registro TRISC. Para configurarlos como entradas: TRISC=0x0003

3.6. Esquema del circuito del procesador

En la figura X se puede observar el circuito del procesador.

Los pines 23, 24, 25 y 26 del dsPIC, que corresponden a AN4, AN5, AN6 y AN7, van conectados a las salidas de los amplificadores AI1, AI2, AI3 y AI4 de circuito de instrumentación.

Los pines correspondientes a V_{dd} y a AV_{dd} van conectados a la alimentación a 3.3V. Los pines correspondientes a V_{ss} y a AV_{ss} van conectados a masa GND. Entre V_{dd} y V_{ss} debe haber un condensador de 100nF, igual que entre AV_{dd} y AV_{ss} .

En CLK1 y CLK0 va conectada la configuración para usar cristal externo que consta del cristal con 2 condensadores de 4,7 μ F conectados a masa en cada extremo. Esta configuración se ha dispuesto en el circuito por si se desea utilizar cristal externo, aunque en nuestro caso al utilizar el oscilador interno no necesitaremos incorporar ningún cristal.

El pin 43 corresponde a U1TX, y el 44 a U1RX, ambos van conectados a los pines 1 y 2 de la regleta de 4 pines P2. En el pin 3 de P2 ira conectada la masa del circuito GND y en el pin 4 la alimentación de 3.3V. Esta regleta se utilizara para conectar el bluetooth aunque también se podría utilizar para conectar un circuito adaptador mediante un MAX3232 para comunicación cableada por RS-232.

La regleta de 5 pines P1 se utilizara para conectar el programador del dsPIC que requiere las siguientes conexiones: en el pin 1 MCLR, en el pin 2 3.3V, en el pin 3 GND, en el pin 4 PGD1, y en el pin 5 PGC1.

Las conexiones realizadas así como los componentes utilizados han seguido las especificaciones determinadas en el datasheet del dsPIC (capítulo “2.0 Guidelines for Getting Started with 16-bit Digital Signal Controllers”).

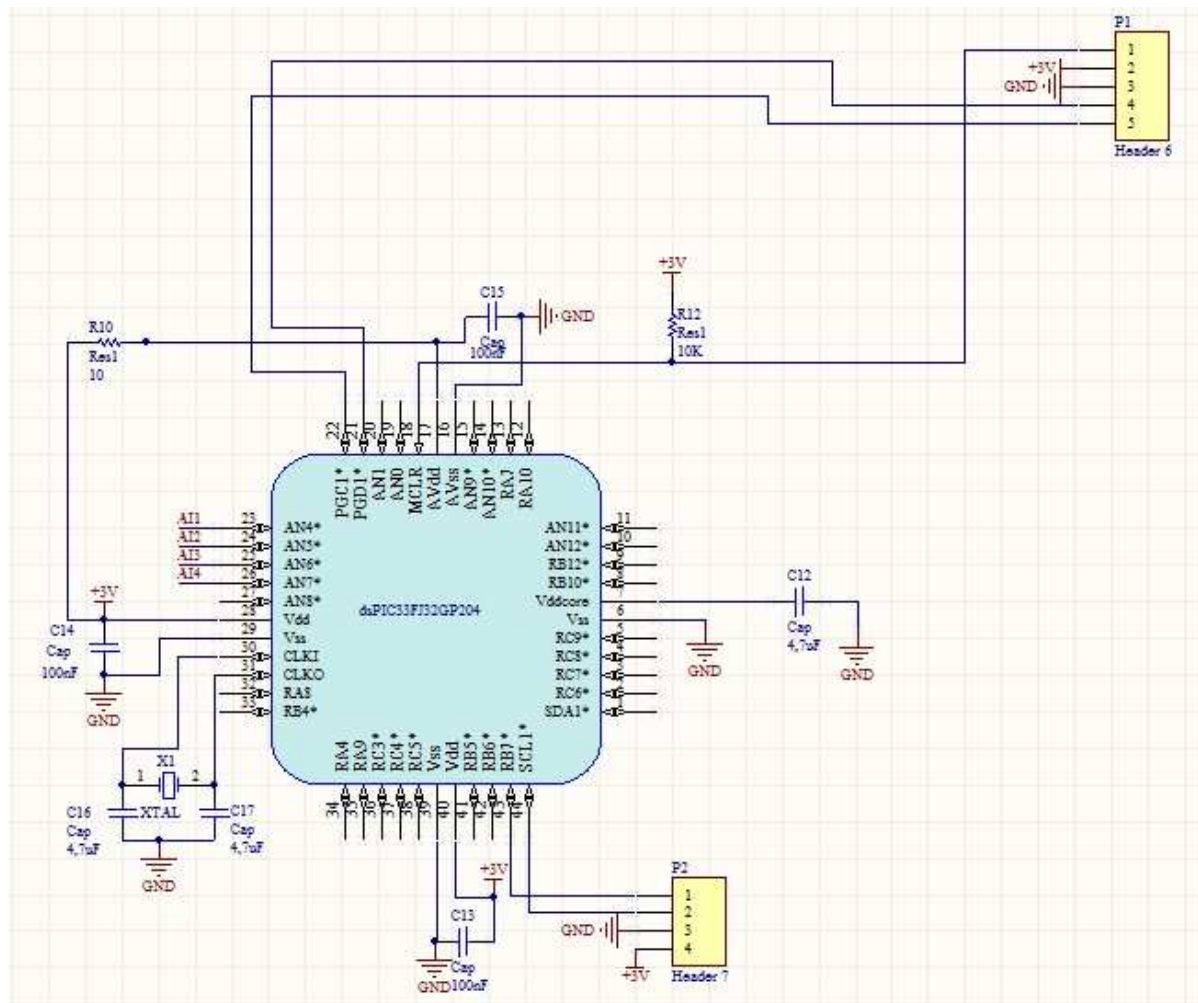


Figura 24 Esquema del circuito del procesador

4.Comunicación con el PC

En este capítulo se define el sistema de comunicación empleado, así como sus características y procedimientos seguidos para su configuración

4. Comunicación con el PC

Como el objetivo final de este proyecto es mostrar los datos (previamente tratados) por pantalla en un ordenador, es necesario el uso de algún dispositivo o conector que nos permita la comunicación entre la plataforma y el ordenador. Por eso, se plantean dos opciones en función del medio por el que se transmiten los datos:

- Utilización de cables conectores (Comunicación cableada)
- Utilización de dispositivos inalámbricos (Comunicación inalámbrica)

Como ya se ha comentado anteriormente, una de las principales características que nos interesa para este proyecto es que sea transportable, y que por tanto la plataforma pueda utilizarse en cualquier entorno que disponga de un ordenador, y, aunque los cables conectores como son USB o DB-9 no interfieren en esta característica sí que limitan la ubicación de la plataforma en el espacio, y hacen de la conexión entre la plataforma y el ordenador una tarea más tediosa.

En cambio, la comunicación inalámbrica favorece la característica de portabilidad y nos permite más libertad a la hora de ubicar la plataforma en el espacio. Es por ello que se considera que la opción más idónea para este proyecto sería la incorporación de un dispositivo inalámbrico para la transmisión y recepción de datos.

Una vez definida la comunicación inalámbrica nos encontramos con diferentes sistemas de comunicación inalámbrica:

- WLAN (Wireless Local Area Network)
- WPAN (Wireless Personal Area Network)

WLAN es una red de área local inalámbrica para la comunicación de datos mediante radiofrecuencia, con un alcance de unos 100 metros aproximadamente, y que permite la comunicación entre dispositivos que se encuentren dentro del área de alcance. Un ejemplo de WLAN es el WI-FI.

WPAN es una red inalámbrica de área personal para la comunicación de datos mediante radiofrecuencia, con un alcance de unos 10 metros aproximadamente, y que permite la comunicación entre dispositivos portátiles personales. Un ejemplo de WPAN es el Bluetooth.

Una de las principales características que diferencian el WI-FI y el Bluetooth es el ancho de banda, la velocidad, la distancia de alcance, así como también el precio, siendo superior para los dispositivos WI-FI.

Dado que en este proyecto la distancia entre la plataforma y el ordenador no será elevada, esta podrá cubrirse con la distancia de alcance de un dispositivo Bluetooth.

4.1. Bluetooth

Bluetooth es una especificación industrial para redes inalámbricas de área personal (WPAN) que permite la transmisión de audio y datos entre diferentes dispositivos a través de un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz (bandas reservadas internacionalmente para uso no comercial de radiofrecuencia electromagnética, en entornos industriales, científicos y médicos).

Esta diseñado especialmente para dispositivos de bajo consumo y con una cobertura baja, características que nos interesan para este proyecto.

En función de la potencia de transmisión los dispositivos Bluetooth se clasifican:

Clase	Potencia máxima permitida (mW)	Potencia máxima permitida (dBm)	Rango aproximado
Clase 1	100mW	20dBm	~100m
Clase 2	2.5mW	4dBm	~25m
Clase 3	1mW	0dBm	~1m

Tabla 3 Clasificación de los dispositivos en función de la potencia de transmisión

También se clasifican en función del ancho de banda:

Versión	Ancho de banda
Versión 1.1	720Kbit/s
Versión 1.2	1Mbit/s
Versión 2.0+EDR	3Mbit/s
Versión 3.0+HS	24Mbit/s

Tabla 4 Clasificación de los dispositivos en función del ancho de banda

EDR (Enhanced Data Rate): Mejora las velocidades de transmisión en hasta 3Mbps.

HS (High Speed): Mayor velocidad, seguridad en las transferencias de datos y mejor administración de energía.

El hardware del dispositivo Bluetooth está formado por:

- Dispositivo de radio encargado de modular y transmitir el señal
- Controlador digital, formado por una CPU, que se encarga de atender las instrucciones relacionadas con el Bluetooth del dispositivo anfitrión, mediante un software denominado Link Manager que se comunica con otros dispositivos a través del protocolo LMP, un procesador de señales digitales (DSP) llamado Link Controller y de las interfaces con el dispositivo anfitrión.

Otro punto importante que caracteriza a un dispositivo Bluetooth es el Perfil Bluetooth:

El Perfil define los servicios que ofrece un dispositivo Bluetooth, por ejemplo: manos libres, dispositivo de interfaz humana, distribución de audio avanzada.

El perfil que nos interesa para este proyecto es el SPP (Serial Port Profile) que mediante el protocolo RFCOMM emula una comunicación por puerto serie RS-232.

4.2. Modulo Promi-ESD02

El modulo Promi-ESD02 es un modulo Bluetooth de clase 2 y v1.1, con antena integrada, entre sus características destacan:

- Alimentación a 3.3V

- Conexión con el dispositivo (dsPIC) vía UART
- Dimensiones reducidas: 18x20x11.7mm
- Fácil instalación del modulo
- Fácil configuración a través del PC mediante RS-232
- Soporta Bluetooth Serial Port Profile (SPP)
- Alcance: 30m



Figura 25 Modulo Bluetooth Promi-ESD02

Como se observa en las características, el modulo es de fácil instalación ya que únicamente cuenta con ocho pines (GND, VDD, STATUS, RST, RXD, TXD, CTS, RTS), de los cuales, si la comunicación se hace sin control de flujo ni reset en el dispositivo, como es nuestro caso, únicamente necesitaríamos conectar los pines GND, VDD, RXD y TXD, y hacer un puente entre CTS y GND.

Es un modulo fácil de utilizar aunque no se dispongan de amplios conocimientos sobre Bluetooth y Radiofrecuencia.

4.2.1. Configuración del módulo

Para la configuración, el modulo incluye una placa a la que se conecta mediante los 8 pines llamada *jig board*, y la cual se conecta mediante un cable serie al PC. También incluye un software para la configuración llamado *Promi-Win*.

Primero hay que colocar el modulo en los pines de la *jig board*, conectar la *jig board* al PC, alimentarla, y finalmente ejecutar el software *Promi-Win* en el PC y seleccionar **PromiWIN configuration**.

Se abre la siguiente pantalla:

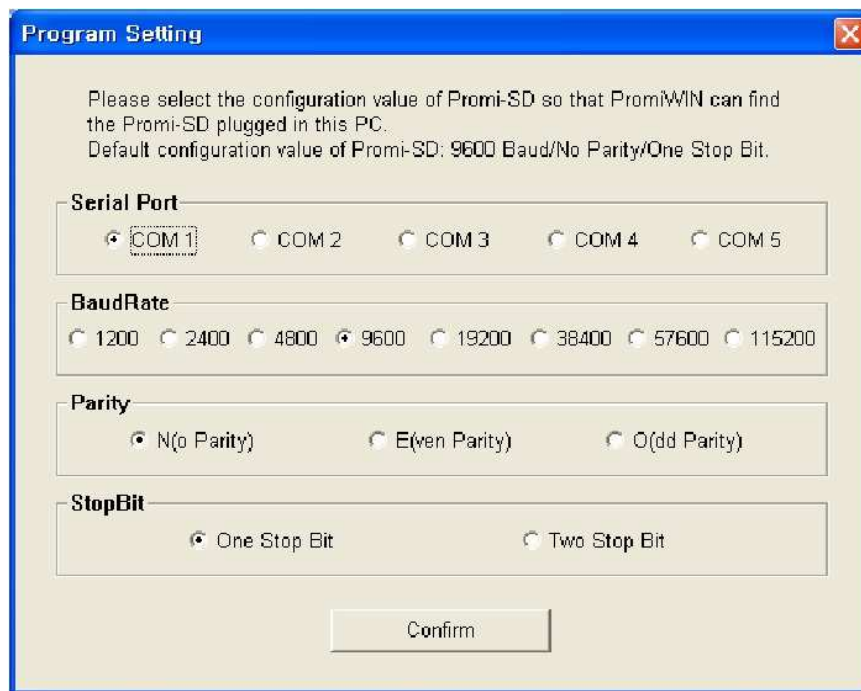


Figura 26 PromiWIN Program Setting

En Serial Port seleccionamos el puerto del PC al que está conectado el cable serie de la *jig board*.

En Baud Rate se selecciona 9600.

En Parity: No parity.

En StopBit: One Stop Bit.

Y hacemos clic en confirm.

Se selecciona Promi-SD>>Start Configuration. Se abrirá la siguiente ventana:

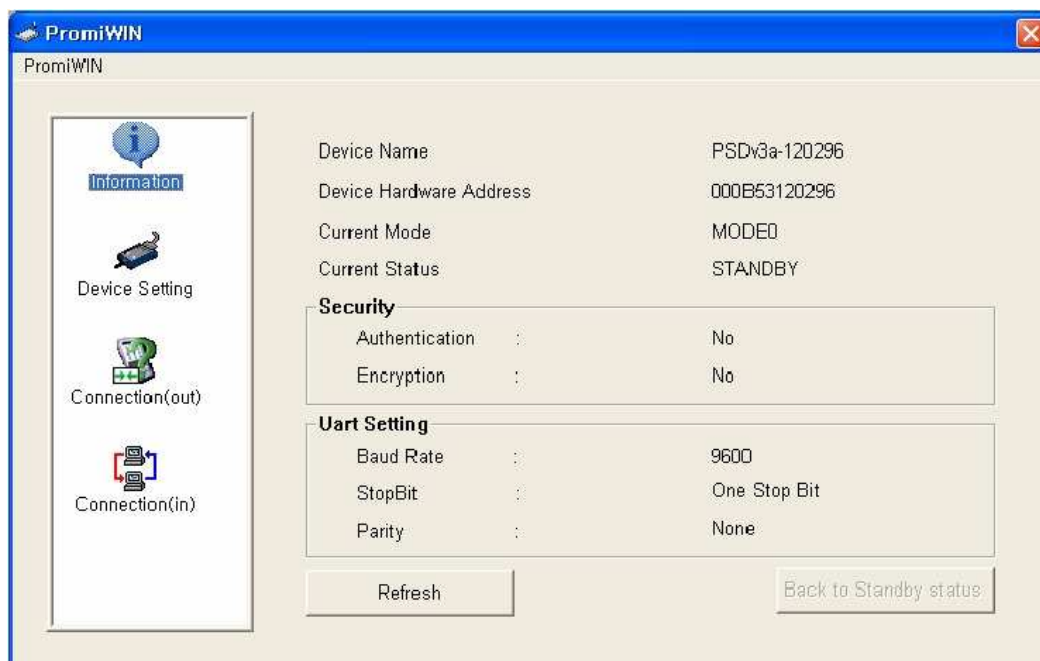


Figura 27 Menú PromiWIN

Seleccionamos Device setting en el menú. Se abrirá la siguiente ventana:

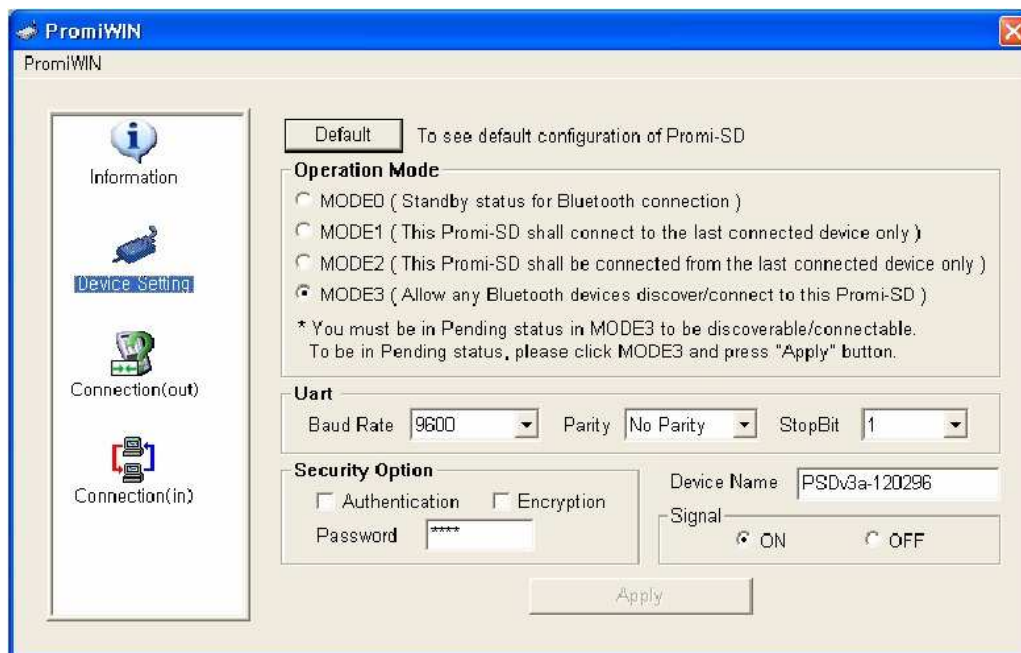


Figura 28 Device Setting

En Operation Mode seleccionamos MODE 3, lo que significa que el dispositivo es visible para cualquier otro cualquier dispositivo.

En UART seleccionamos la misma configuración utilizada en la UART1 de dsPIC, que es: 9600, No Parity y 1 StopBit.

En Device Name podemos cambiar el nombre del dispositivo cuando está en modo visible. En este caso el dispositivo tendrá el siguiente nombre: CETpD_2.

En security Option se selecciona autenticación mediante el Password: 1234.

Hacemos clic en Apply y quedará configurado como dispositivo Bluetooth entrante y ya estará listo para funcionar.

A partir de este momento el dispositivo Bluetooth está listo para que otro dispositivo Bluetooth saliente inicie la conexión. Una vez realizada la conexión, ambos dispositivos se comportan como puertos serie tradicionales como si estuvieran conectados por cable ya que la comunicación inalámbrica se realiza de forma transparente al utilizar el protocolo RFCOMM del conjunto de servicios Bluetooth.

4.3. Emparejamiento de los dispositivos Bluetooth

Para que la comunicación mediante entre la PCB y el ordenador se deben realizar unos sencillos pasos que nos permitirán emparejar el modulo bluetooth de la PCB con el dispositivo USB bluetooth del ordenador.

A continuación se detallan los pasos a seguir para emparejar dos dispositivos utilizando Windows 7.

Primeramente se conecta el dispositivo USB Bluetooth al ordenador y se espera a que Windows realice la instalación de los drivers necesarios. Una vez finalizada la instalación se muestra en el lateral izquierdo de la barra de tareas de Windows el símbolo Bluetooth. Se alimenta la PCB y haciendo clic derecho sobre el icono Bluetooth seleccionamos Agregar un dispositivo, tal y como muestra la Figura 29

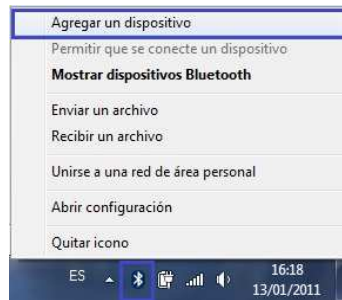


Figura 29 Agregar dispositivo Bluetooth

Se abrirá una ventana en la cual aparecerán los dispositivos Bluetooth visibles (Figura 30). Seleccionaremos el dispositivo CETpD_2, que corresponde al modulo Promi-ESD02, y se hace clic en continuar.

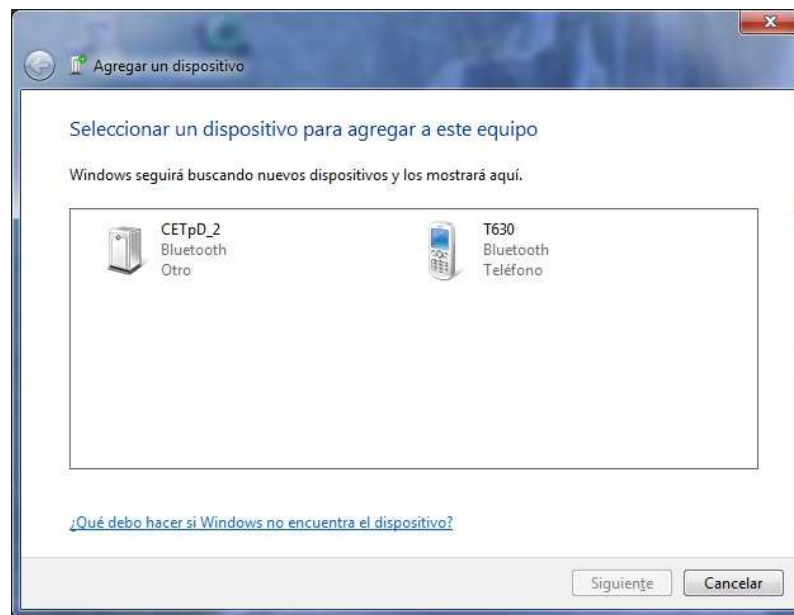


Figura 30 Selección del dispositivo Bluetooth

En la siguiente ventana debemos seleccionar la opción de emparejamiento entre crear un código propio de emparejamiento, escribir el código de emparejamiento del dispositivo o aparear sin usar códigos. Como se ha visto en la configuración del modulo Bluetooth Promi-ESD02, el dispositivo está configurado con un código de autenticación, por lo que se selecciona la segunda opción: Escribir el código de emparejamiento del dispositivo (Figura 31).

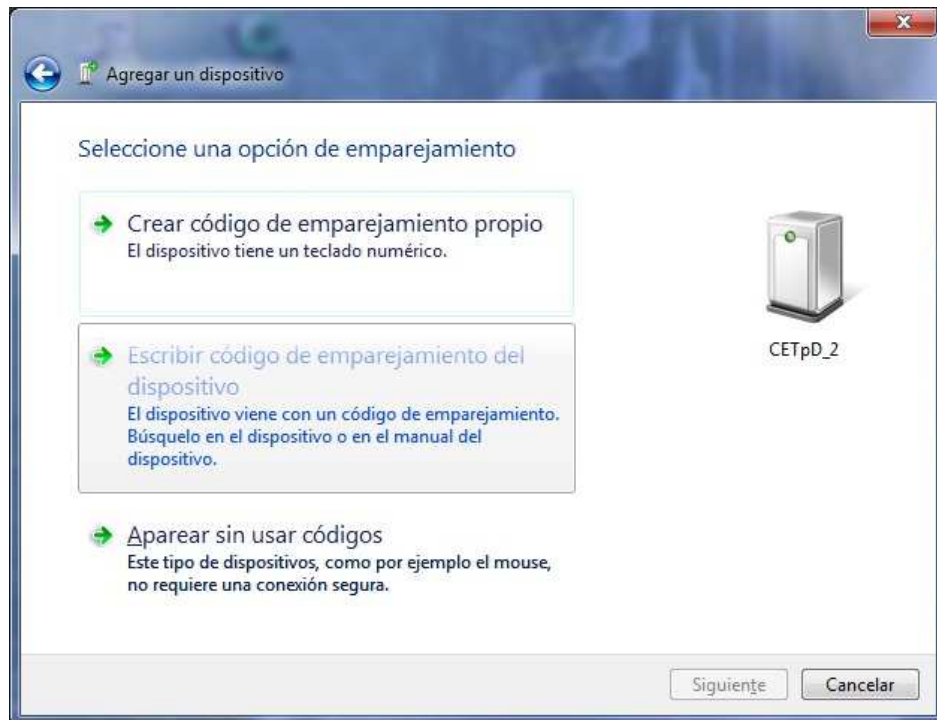


Figura 31 Selección de la opción de emparejamiento

En la siguiente ventana se deberá escribir el código de emparejamiento, que corresponde con el Password de autenticación del dispositivo Promi-ESD02 (1234), tal y como se muestra en la Figura 32

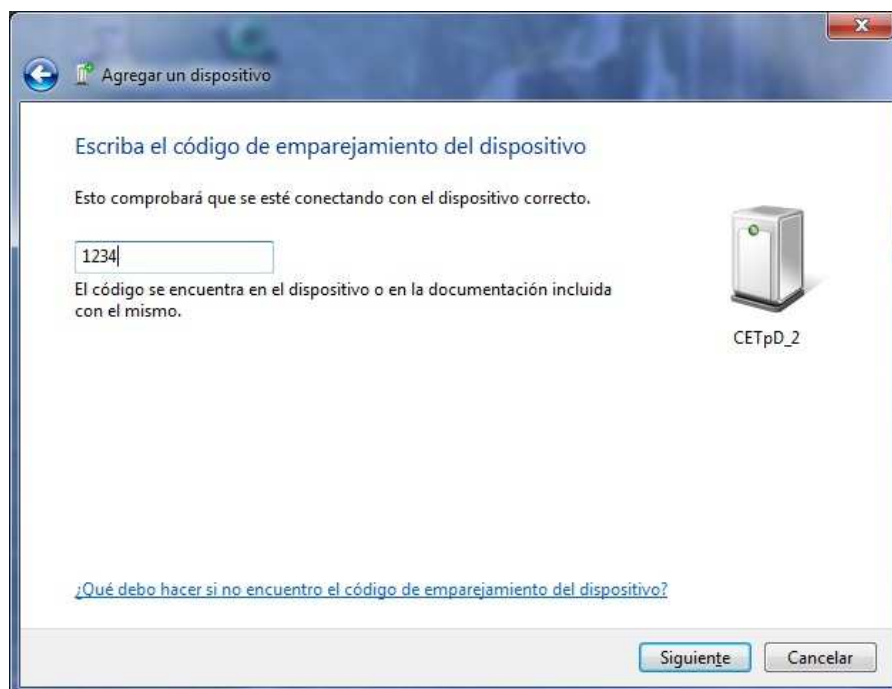


Figura 32 Introducción del código de emparejamiento del dispositivo

En la siguiente ventana nos indicara si el dispositivo se ha agregado correctamente:

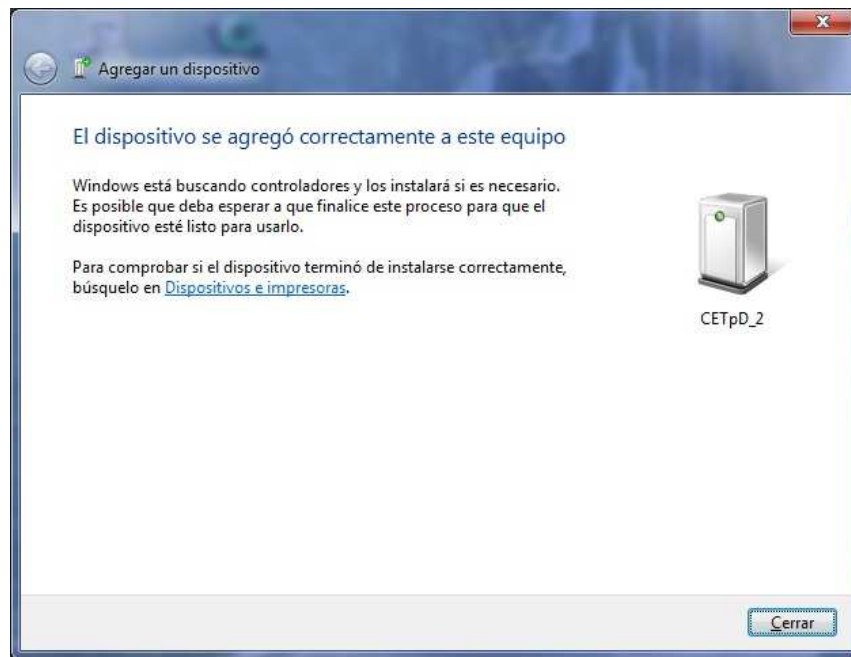


Figura 33 Dispositivo agregado

Una vez agregado el dispositivo se debe saber a qué puerto COM está asociado para así poder iniciar la comunicación. Para saber el puerto COM asociado se hace clic derecho sobre el icono Bluetooth de la barra de tareas de Windows y se selecciona Abrir configuración. Se abrirá la ventana que se muestra a continuación:

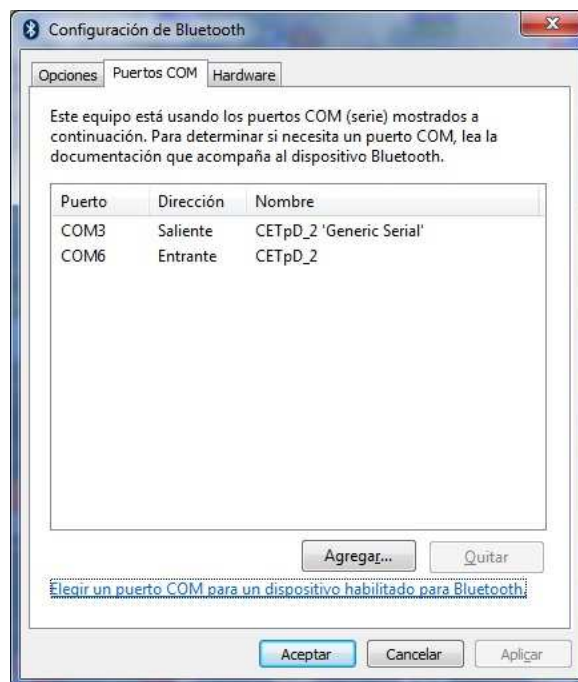


Figura 34 Configuración Bluetooth

En la pestaña Puertos COM aparecerán los dos puertos asociados al dispositivo. Uno como entrante y otro como saliente. Nos interesa saber el puerto COM saliente, que es mediante el cual el ordenador se conecta con el dispositivo funcionando como un puerto serie.

Para asegurarnos de que los datos se reciben al ordenador correctamente se visualizaran mediante la herramienta Serial Port Monitor del Software PIC C Compiler. Donde primero habrá que configurar el puerto por el que se realizara la comunicación, para ello seleccionamos Configuration>>Set Port Options.

Se abrirá una ventana de configuración como la que se muestra en la Figura 35:

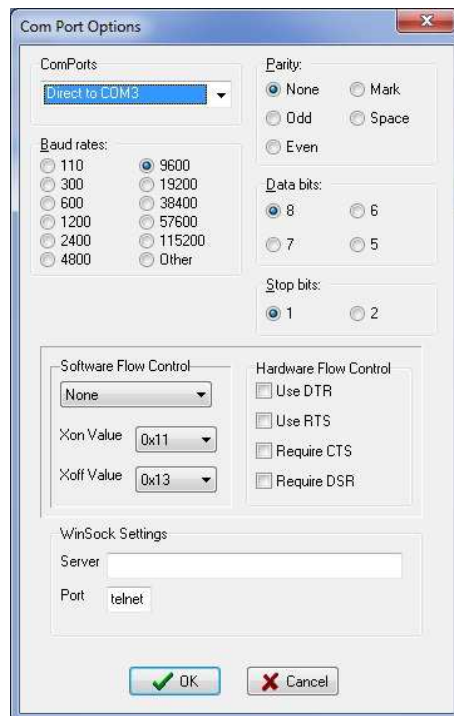


Figura 35 Configuración del puerto COM

En ComPorts se debe seleccionar el puerto COM saliente asociado al dispositivo, como se ha visto antes, en este caso será el puerto COM3. El resto de la configuración debe corresponderse con la configuración de la UART y del dispositivo Bluetooth, es decir: Sin paridad, datos de 8 bits, Baud Rate=9600 y sin control de flujo. Una vez configurado hacemos clic en OK.

También se deben configurar las opciones de visualización de los datos ASCII seleccionando Configuration>>ASCII Options>>Display Extended ASCII.

A partir de este momento ya se pueden visualizar correctamente los datos recibidos haciendo clic en View (Figura 36).

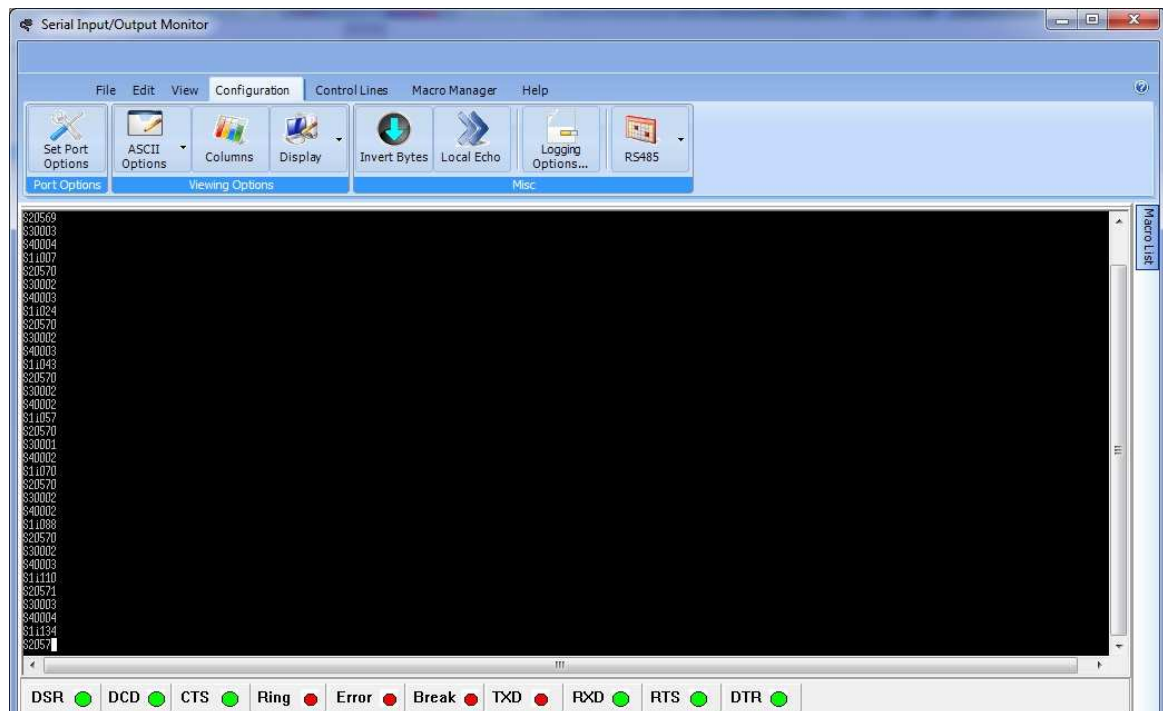


Figura 36 Visualización de los datos a través del Serial Port Monitor

5.Elaboración y diseño de la PCB

En este capítulo se define el concepto de PCB así como el software utilizado para el diseño, el método de elaboración y los resultados obtenidos.

5. Elaboración y diseño de la PCB

En esta sección se define el concepto de PCB y se detalla el software utilizado así como los pasos seguidos para la elaboración y el diseño de la PCB de este proyecto.

5.1. PCB

Una PCB, placa de circuito impreso, (Figura 37) consiste en una placa formada por hojas de cobre sobre un sustrato no conductor, como puede ser la baquelita o fibra de vidrio, sobre la que se colocan los componentes electrónicos, los cuales estarán conectados eléctricamente mediante la grabación de pistas sobre la placa conductora.

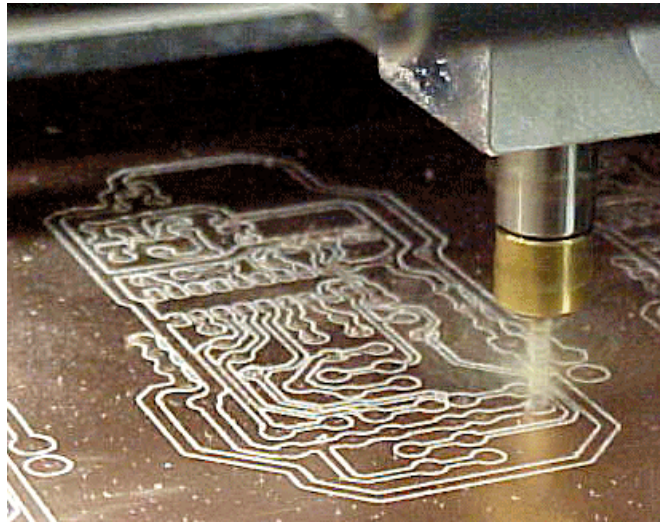
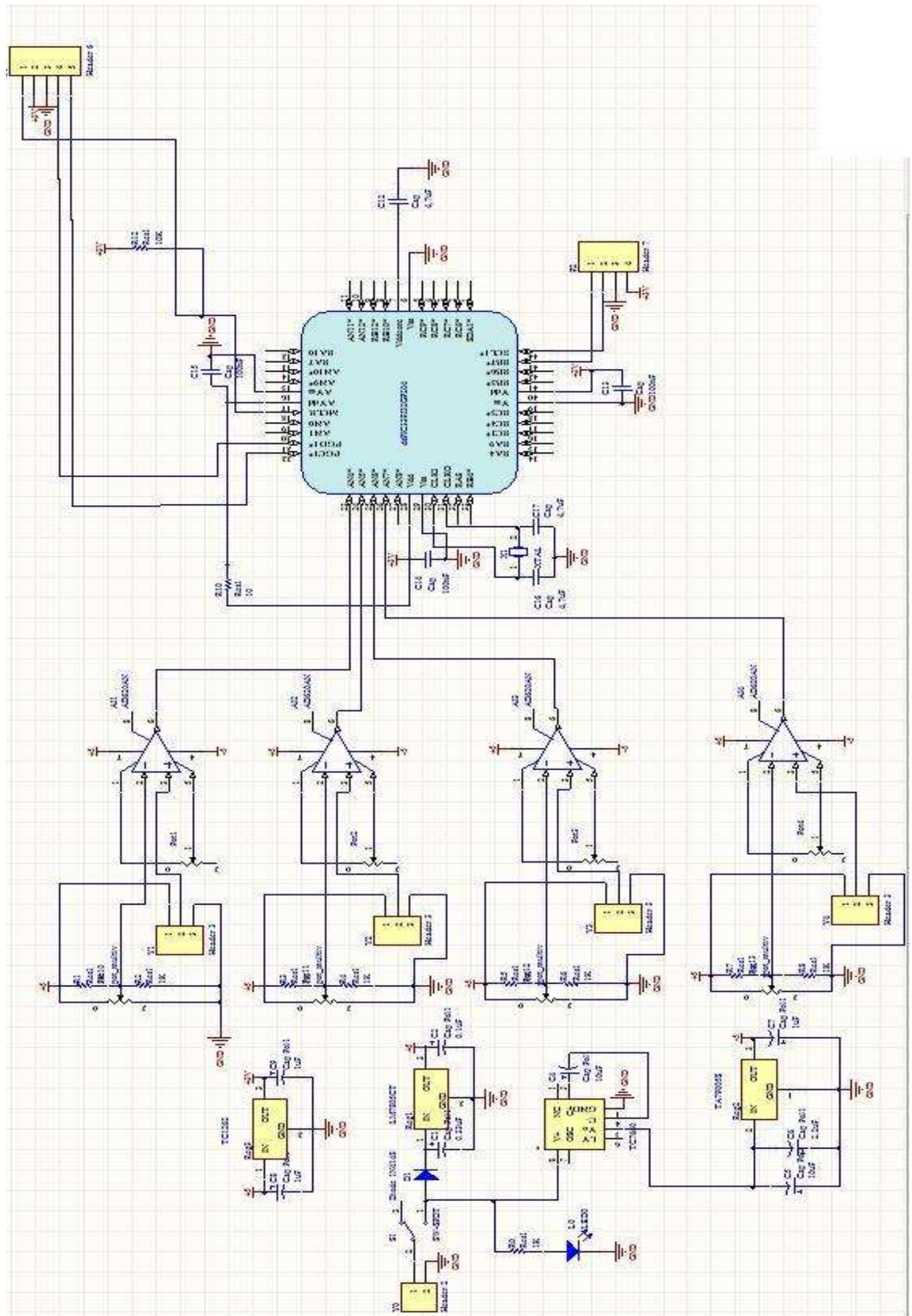


Figura 37 Placa de circuito impreso

5.2. Diseño del circuito impreso

Para el diseño del circuito impreso se ha utilizado el software Altium Designer Summer, las características del software así como los pasos básicos para crear un diseño de PCB se exponen en el “Anexo 1: Diseño de una PCB en Altium Designer”.

A continuación se muestra el esquemático del circuito del proyecto:



Y el diseño de la PCB a doble capa del circuito del proyecto:

- Top Layer:

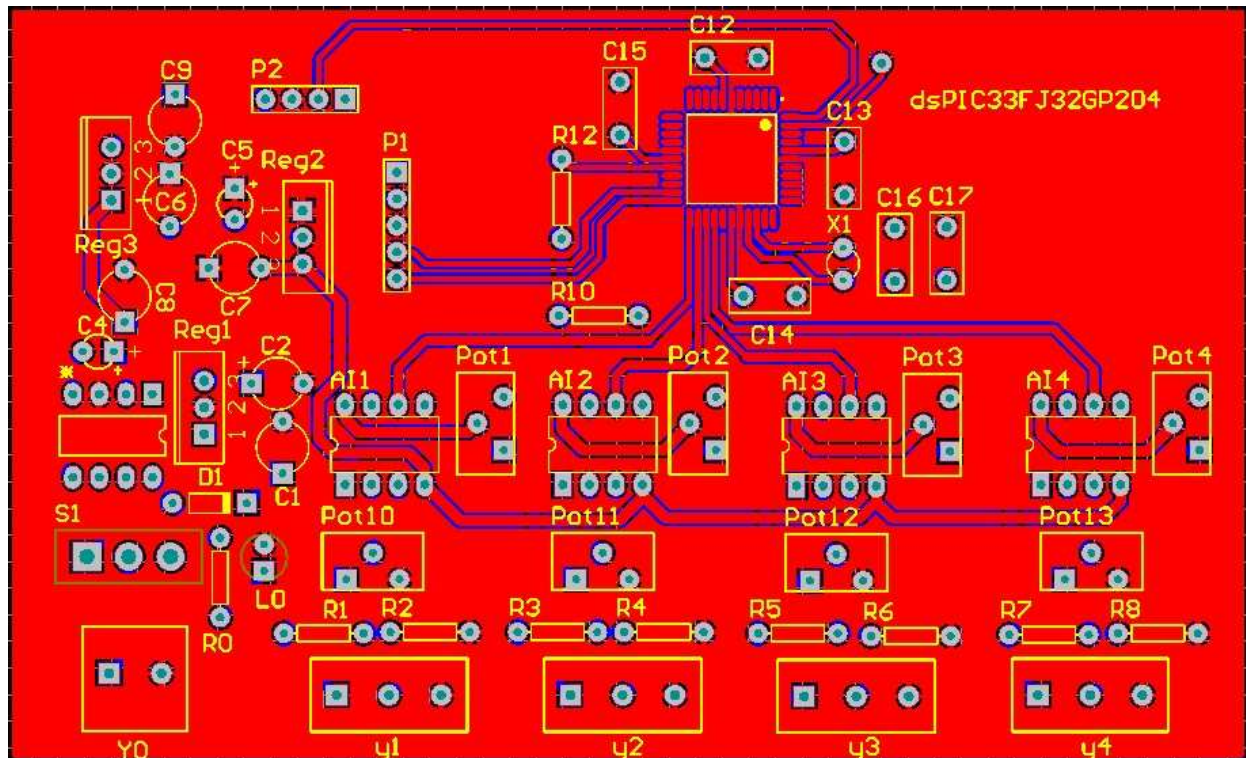


Figura 38 Vista de la Top Layer de la PCB

- Bottom Layer:

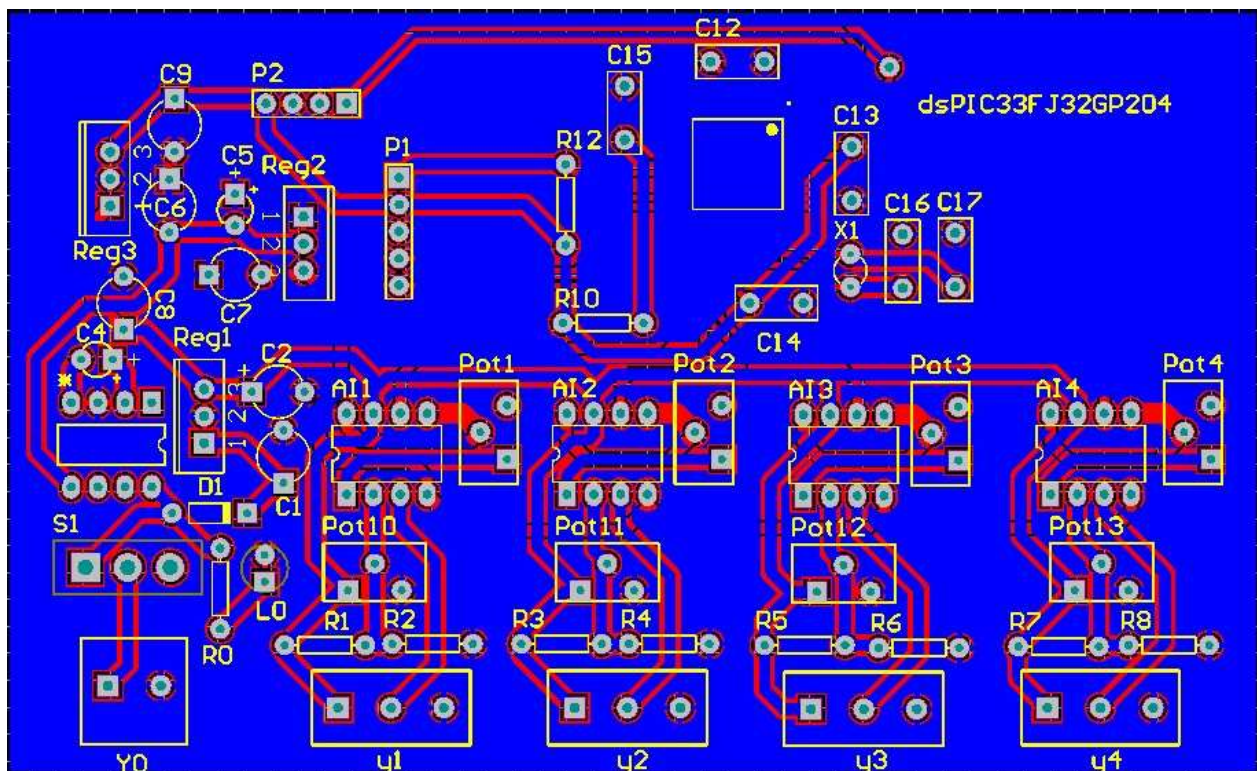


Figura 39 Vista de la Bottom Layer de la PCB

5.3. Métodos para la elaboración de circuitos impresos

La gran mayoría de las placas para circuitos impresos se hacen retirando el cobre no deseado, dejando sólo las pistas de cobre deseado.

Existen diferentes métodos para la producción de circuitos impresos:

- La impresión serigráfica utiliza tintas resistentes al grabado para proteger la capa de cobre. Los grabados posteriores retiran el cobre no deseado. Alternativamente, la tinta puede ser conductiva, y se imprime en una tarjeta virgen no conductiva.
- El fotograbado utiliza una fotomecánica y grabado químico para eliminar la capa de cobre del sustrato. La fotomecánica usualmente se prepara con un fotoplotter, a partir de los datos producidos por un programa para el diseño de circuitos impresos. También se utilizan transparencias impresas en una impresora Láser como fotoherramientas de baja resolución.
- El fresado de circuitos impresos utiliza una fresa mecánica de 2 o 3 ejes para quitar el cobre del sustrato. Una fresa para circuitos impresos funciona en forma similar a un plotter, recibiendo comandos desde un programa que controla el cabezal de la fresa los ejes x, y y z. Los datos para controlar la máquina son generados por el programa de diseño, y son almacenados en un archivo en formato HPGL o Gerber.
- La impresión en material termosensible para transferir a través de calor a la placa de cobre.

Tanto el recubrimiento con tinta, como el fotograbado requieren de un proceso de atacado químico, en el cual el cobre excedente es eliminado, quedando únicamente el patrón deseado.

Para este proyecto el método utilizado ha sido el fresado del circuito impreso mediante una fresadora Proto C30 del fabricante LPKF.

Una vez se ha realizado la PCB con la fresadora, únicamente se tiene que colocar y soldar los componentes en la ubicación que les corresponda ya que la fresadora ProtoMat se encarga de hacer todos los agujeros. Posteriormente a la soldadura de los componentes la PCB presenta la apariencia de la Figura 40.

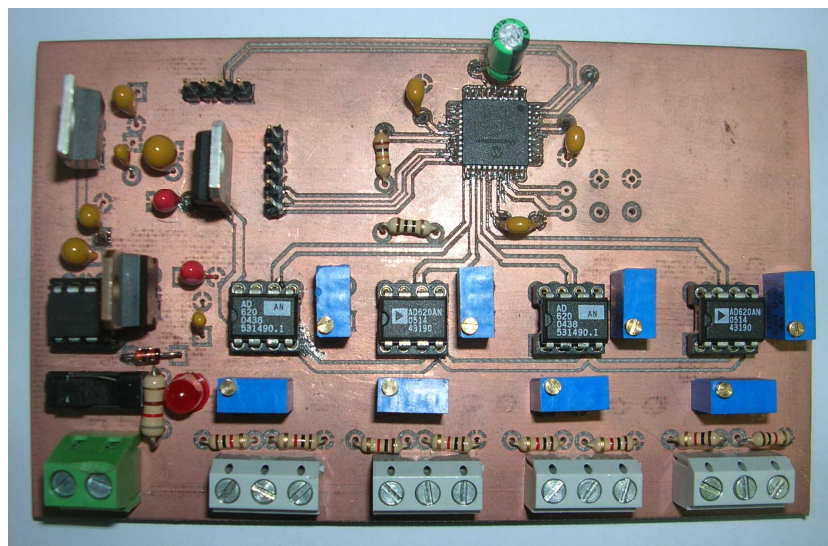


Figura 40 PCB montada

6.Programas realizados

En este capítulo se hace referencia al Hardware del proyecto compuesto por el programa del microcontrolador y la aplicación de ejemplo para el ordenador.

6. Programas realizados

6.1. Programa dsPIC

Este subcapítulo contiene una explicación detallada del programa del dsPIC incluyendo un diagrama de flujo y el código en C elaborado para este proyecto.

Para la programación del dsPIC se ha utilizado el software MPLAB de Microchip que nos permite realizar tareas como la creación y edición del código del programa, simulación, programación, depuración y nos permite seleccionar el compilador. Para este proyecto se utiliza el compilador MPLAB C30, que convierte programas elaborados en código C (estándar ANSI C) a lenguaje ensamblador.

6.1.1. Explicación de programa

A continuación se muestra un resumen del programa a través de la explicación de cada una de las funciones empleadas, interrupciones y del programa principal.

6.1.1.1. Programa Principal

Al inicio se configurarán el oscilador, el ADC y la UART mediante la llamada a las funciones:

```
init_OSC();
```

```
init_ADC();
```

```
init_UART();
```

Se activa el modulo ADC, el cual irá muestreando y convirtiendo los datos de las cuatro entradas analógicas, y almacenará y tratará los valores obtenidos en la **IRQ del ADC1**.

Cuando se hayan realizado 10 conversiones para cada sensor, se desactiva la interrupción de conversión del modulo ADC y la variable media toma el valor 1, lo que nos permitirá realizar la media de los valores de cada uno de los sensores ($s[0]$, $s[1]$, $s[2]$ y $s[3]$).

Una vez calculada la media de cada sensor, se almacenaran en la tabla $t[34]$ los valores en ASCII de cada sensor precedidos por el correspondiente identificador (S1, S2, S3 o S4) y seguidos por un final de línea (LF) y retorno del carro (CR). Para ello se utilizara una tabla de 32 posiciones $t[34]$, en las dos posiciones iniciales se introducirá un final de línea (LF) y un retorno del carro (CR), las posiciones restantes se llenaran correspondiendo 8 posiciones para cada sensor, que contienen la siguiente información:

1. Valor 0x53, correspondiente a la S mayúscula en ASCII.
2. Numero identificador del sensor (1, 2, 3 o 4) en ASCII (0x31, 0x32, 0x33 o 0x34).
3. Primer dígito ASCII del valor del sensor.
4. Segundo dígito ASCII del valor del sensor.
5. Tercer dígito ASCII del valor del sensor.
6. Cuarto dígito ASCII del valor del sensor.
7. Valor 0x0D correspondiente al retorno del carro (CR) en ASCII
8. Valor 0x0A correspondiente a nueva línea (LF) en ASCII

Una vez almacenados los valores en $t[34]$ se procederá a la transmisión de estos activando la transmisión (*UTXEN*) y llamando a la función **transmit_t(t)**, que transmitirá los valores de $t[34]$.

Cuando se hayan transmitido por la UART todos los valores de la tabla se desactivara la transmisión (*UTXEN*) y el programa volverá a la parte en que se activa la interrupción de conversión del ADC.

6.1.1.2. Funciones

- **Función init_OSC**

Esta función configura el oscilador como oscilador interno FRC con PLL para obtener una frecuencia de operación de 40MIPS y espera a que se produzca el cambio y se fije el PLL.

- **Función init_ADC**

Esta función configura el ADC para que funcione en modo 1canal-12bits mediante el Channel 0 del adc, realice un escaneo de las entradas analógicas AN4, AN5, AN6 y AN7 tomando como tensiones de referencia la tensión de alimentación (3.3V y GND), muestreo y conversión automático y interrupción en la finalización de la secuencia de muestreo-conversión de las 4 entradas.

Esta función configura los registros de acuerdo con lo expuesto en el apartado “3.2 Configuración del conversor AD”.

- **Función init_UART**

Esta función configura la UART1 en modo 8 bits, 1 bit de stop y sin paridad, utilice los pines U1TX y U1RX, Baud Rate a 9600 y se produzca una interrupción cuando se transmita un carácter y el buffer de transmisión U1TXREG esté vacío.

Una vez configurada se activa la UART (*UARTEN*).

Esta función configura los registros de acuerdo con lo expuesto en el apartado “3.3 Configuración de la UART”.

- **Función transmit_t**

Esta función irá enviando uno a uno, hasta 34, los valores de la tabla *t[34]* a la función **transmit**.

- **Función transmit**

En esta función esperamos hasta que el buffer de transmisión este vacío mediante el uso de la variable *t_flag*, la cual toma el valor 1 cuando se produce la **IRQ de U1TX**.

Cuando el buffer de transmisión está vacío (*t_flag* =1) se resetea la variable *t_flag* ya que se procede a introducir en el buffer de transmisión el valor correspondiente de la tabla *t[34]*.

6.1.1.3. Interrupciones

- **IRQ del ADC1**

Cuando se produce esta interrupción los valores obtenidos de la conversión de AN4, AN5, AN6 y AN7, que se encuentran en ADC1BUF0, ADC1BUF1, ADC1BUF2 y ADC1BUF3, se suman a los valores obtenidos de la conversión anterior y se almacenan en la misma variable *s[0]*, *s[1]*, *s[2]* y *s[3]*.

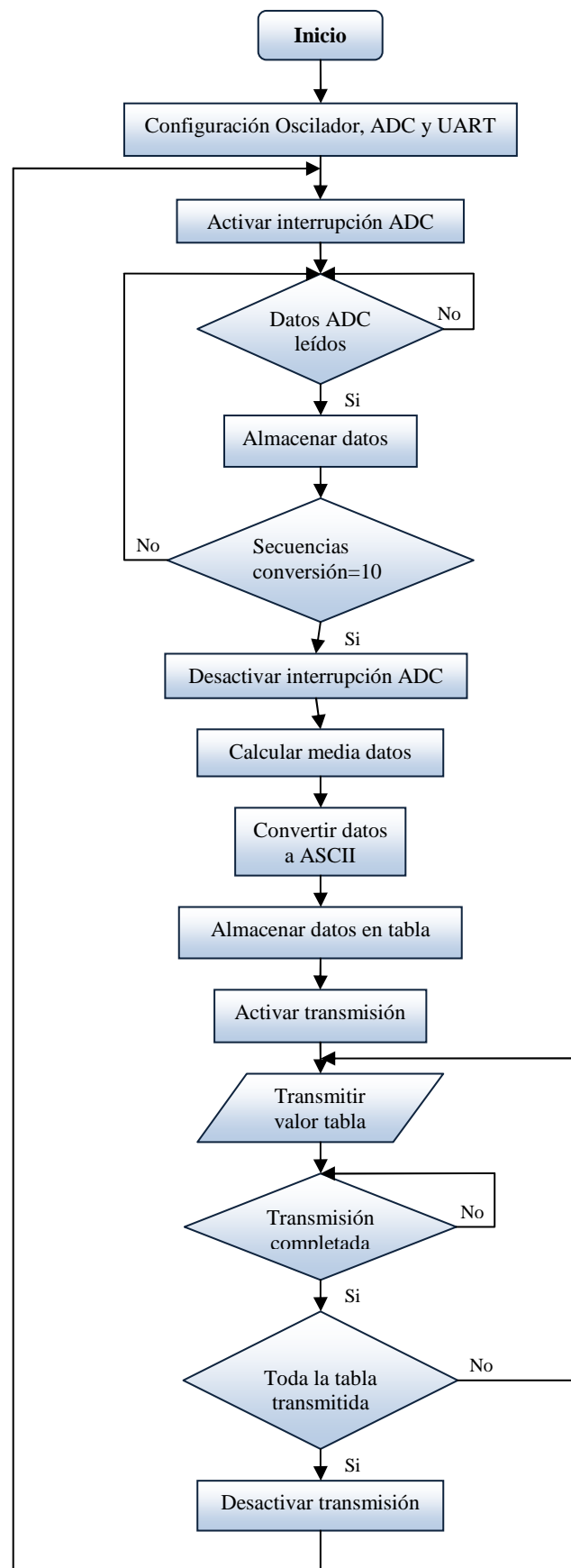
Se comprueba mediante el contador *i* si se han realizado 10 secuencias de muestreo-conversión de las 4 entradas, si es así la variable *media* toma el valor 1, lo que indica que ya se puede proceder a calcular la media de los 10 valores de cada sensor y se desactiva la interrupción de conversión del ADC; si no es así, se incrementa el contador.

Finalmente se resetea el flag de interrupción.

- **IRQ de U1TX**

Esta interrupción indica que el buffer de transmisión esta vacío por lo que se asigna el valor 1 a *t_flag*, que permitirá introducir el siguiente valor a transmitir en la **función transmit**.

6.1.2. Diagrama de flujo del programa

**Figura 41** Diagrama de flujo del programa

6.1.3. Programa dsPIC

A continuación se muestra el programa en código C, argumentando cada instrucción:

```
#include "p33FJ32GP204.h"
```

```
_FOSCSEL(FNOSC_FRCPLL); Configuración de los registros FOSCSEL Y FOSC detallados en el punto "1.1  
_FOSC(FCKSM_CSDCMD & OSCIOFNC_OFF); Configuración del oscilador"
```

La declaración de las cuatros siguientes variables de tipo volatile es debido a que dichas variables son modificadas en interrupciones del programa

```
volatile unsigned int s[4]; Tabla para almacenar los valores de los 4 sensores convertidos en el ADC.
```

```
volatile unsigned int t_flag; Variable que indicará si se puede realizar la siguiente transmisión.
```

```
volatile unsigned int media; Variable que nos indicará si se puede hacer la media.
```

```
volatile unsigned int i; Variable que indicará si se han obtenido 10 valores (10 muestro-conversión)de cada sensor.
```

```
void init_OSC(){ Función para la configuración del ADC
```

```
    PLLFBD=41; Valor M-2 para la configuración del PLL
```

```
    CLKDIVbits.PLLPOST=0; Valor N2-2 para la configuración del PLL
```

```
    CLKDIVbits.PLLPRE=0; Valor N1-2 para la configuración del PLL
```

```
    __builtin_write_OSCCONH(0x01); Iniciar el Clock Switch como FRC interno con PLL
```

```
    __builtin_write_OSCCONL(0x01);
```

```
    while(OSCCONbits.COSC!=0b001); Esperar a que ocurra el Clock Switch
```

```
    while(OSCCONbits.LOCK!=1){ }; Esperar a que se fije el PLL
```

```
}
```

```
void init_ADC(){
```

```
    AD1PCFGLbits.PCFG4=0; Configuración de los pines AN4, AN5, AN6 y AN7 como entradas analógicas
```

```
    AD1PCFGLbits.PCFG5=0;
```

```
    AD1PCFGLbits.PCFG6=0;
```

```
    AD1PCFGLbits.PCFG7=0;
```

```
    AD1CON1=0x04E4; Configuración de los registros AD1CON1, AD1CON2, AD1CON3 y AD1CSSL
```

```
    AD1CON1bits.AD12B=1; detallados en el punto "1.2 Configuración del conversor AD"
```

```
    AD1CON2=0x040C;
```

```
    AD1CON3=0x8E00;
```

```
    AD1CSSL=0x00F0;
```

```
    IFS0bits.AD1IF=0; Se resetea el flag de interrupción de conversión completa del ADC1
```

```
}
```

```

void init_UART(){
    U1BRG=0x0103; Configuración para Baud Rate=9600
    U1MODE=0x0000; Configuración de los registros U1MODE y U1STA detallados en el punto "1.3
    U1STA=0x8000; Configuración de la UART"
    IPC7=0x4400;
    IFS0bits.U1TXIF=0; Se resetea el flag de interrupción de transmisión
    IFS0bits.U1RXIF=0; Se resetea el flag de interrupción de recepción
    IEC0bits.U1TXIE=1; Se activa la interrupción de transmisión
    IEC0bits.U1RXIE=1; Se activa la interrupción de recepción
    U1MODEbits.UARTEN=1; Se activa la UART1
}

void __attribute__((interrupt,no_auto_psv)) _ADC1Interrupt(void){ IRQ del ADC1
    s[0]=s[0]+ADC1BUF0; Se guarda en s[0] la suma de los valores de cada conversión de AN4
    s[1]=s[1]+ADC1BUF1; Se guarda en s[1] la suma de los valores de cada conversión de AN5
    s[2]=s[2]+ADC1BUF2; Se guarda en s[2] la suma de los valores de cada conversión de AN6
    s[3]=s[3]+ADC1BUF3; Se guarda en s[3] la suma de los valores de cada conversión de AN7
    if(i==9){ Si el contador del numero de conversiones (i) es 10 activamos el flag media, que indicara que ya se
        media=1; puede hacer la media de los 10 valores de cada sensor y desactivamos la interrupción del
        IEC0bits.AD1IE=0; modulo ADC.
    }
    else{ Sino, aumentamos en uno el valor del contador de conversiones
        i++;
    }
    IFS0bits.AD1IF=0; Se resetea el flag de interrupción de conversión completa del ADC1
}

void __attribute__((interrupt,no_auto_psv)) _U1TXInterrupt(void){ IRQ de U1TX
    t_flag=1; Se activara t_flag que utilizaremos para indicar que se ha completado la transmisión de 8 bits
    IFS0bits.U1TXIF=0; Se resetea el flag de interrupción de transmisión
}

void transmit(char tx){ función que escribe el valor tx en el buffer de transmisión cuando éste está vacío
    while(t_flag==0); Espera hasta que se hayan transmitido los 8 bits anteriores
    t_flag=0; Se desactiva la t_flag para indicar que hay 8 bits en proceso de transmisión
    U1TXREG=tx; Escribe en el buffer de transmisión los siguientes 8 bits
}

```

```

void transmit_t(char *vec){ función que va enviando los 34 valores de la tabla a transmitir
    int y=0; Se inicializa el contador de posiciones de la tabla en 0
    while(y<34){ Se van enviando los 34 valores de la tabla a la función transmit
        transmit(vec[y]);
        y++;
    }
}

```

```

int main(void){
    RCONbits.SWDTEN=0; Se desactiva el Watchdog timer
    TRISB=0x000C; Configuración de los pines 23, 24, 25 y 26 como entradas
    TRISC=0x0003;
    RPINR18bits.U1RXR=0b01000; Configuración de los pines remapables 43 y 44 como UITX y U1RX
    RPOR3bits.RP7R=0b00011;

    init_OSC(); Llamada a la función de configuración del oscilador
    init_ADC(); Llamada a la función de configuración del ADC
    init_UART(); Llamada a la función de configuración de la UART

    int j, n; Declaramos 2 variables enteras, j para moverse en la tabla de caracteres ASCII, y n para moverse en
        la tabla que contiene los valores de los 4 sensores
    char t[34]; Declaramos la tabla t que contendrá los 34 valores a transmitir
    unsigned int aux; Variable auxiliar para cálculos
    t_flag=0; Se resetea el flag que se utilizara para indicar que se han transmitido 8 bits
    AD1CON1bits.ADON=1; Se activa el conversor AD

    while(1){ Esta parte del programa siempre esta ejecutándose
        IEC0bits.AD1IE=1; Se activa la interrupción de conversión completa del ADC1
        if(media==1){ Cuando se hayan convertido 10 valores para cada uno de los 4 sensores en la IRQ
            del ADC media=1
            i=0; Se resetea el contador del numero de conversiones
            s[0]=s[0]/10; Se hace la media para cada uno de los 4 sensores dividiendo entre 10 la suma
            s[1]=s[1]/10; de los diez valores de cada uno
            s[2]=s[2]/10;
            s[3]=s[3]/10;

```

```

t[0]=0x0D; Se introduce en las dos primeras posiciones de la tabla los valores en ASCII
t[1]=0x0A; correspondientes a un retorno del carro (CR) y nueva línea de línea(LF)
j=2; Inicializamos j en 2 que se utilizara como índice para desplazarse por la tabla
for(n=0;n<4;n++){ Se convertirá el valor de cada sensor a ASCII
    t[j]=0x53; En esta ubicación se introduce la S en ASCII
    j++;
    t[j]=n+0x31; En esta ubicación se introduce el numero del sensor en ASCII
    j++;
    t[j]=(s[n]/1000)+0x30; En esta ubicación se introduce el primer digito ASCII
                           del correspondiente sensor
    j++;
    aux=s[n]%1000;
    t[j]=(aux/100)+0x30; En esta ubicación se introduce el segundo digito ASCII
                           del correspondiente sensor
    j++;
    aux=aux%100;
    t[j]=(aux/10)+0x30; En esta ubicación se introduce el tercer digito ASCII
                           del correspondiente sensor
    j++;
    t[j]=(aux%10)+0x30; En esta ubicación se introduce el cuarto digito ASCII
                           del correspondiente sensor
    j++;
    t[j]=0x0D; En esta ubicación se introduce el valor ASCII correspondiente al
                           retorno del carro (CR)
    j++;
    t[j]=0x0A; En esta ubicación se introduce el valor ASCII correspondiente a
                           nueva línea (LF)
    j++;
}
U1STAbits.UTXEN=1; Se activa la transmisión
transmit_t(t); Se llama a la función transmit_t que gestionara la transmisión de los valores
               de la tabla
U1STAbits.UTXEN=0; Se desactiva la transmisión
s[0]=0; Se resetean los valores de los sensores almacenados en s[4]
s[1]=0;
s[2]=0;
s[3]=0;
}

```

```

    }
}

```

6.2. Aplicación de ejemplo

Para el tratamiento y visualización de los datos en el ordenador se ha desarrollado una pequeña aplicación como ejemplo utilizando LabView. La aplicación se conectará mediante bluetooth con el sistema de posturografía y comenzará a recibir los datos de los sensores, la aplicación tratará los datos y los mostrara mediante un grafico XY que muestra en todo momento la posición del centro de gravedad.

Para el tratamiento de los datos hay que definir primeramente los siguientes conceptos:

La fuerza total ejercida sobre la plataforma se calcula mediante la suma de las fuerzas ejercidas sobre cada uno de los 4 sensores:

$$F_t = S1 + S2 + S3 + S4$$

Para el cálculo de la posición del centro de gravedad en los ejes XY, se parte de la base que el centro de la plataforma se corresponde con el origen de los ejes XY, como se muestra en la Figura 42.

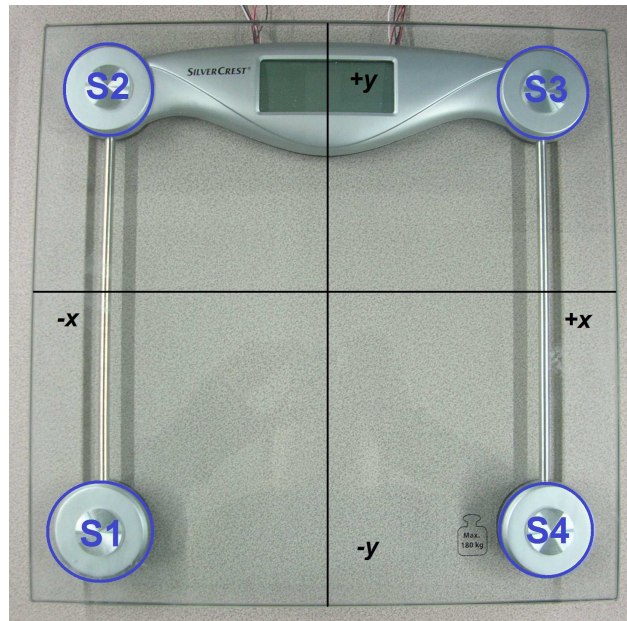


Figura 42 Situación de los ejes XY en la plataforma

Entonces, para el cálculo de F_x , los sensores situados a la izquierda se consideran valores negativos de x, y los situados a la derecha se consideran positivos, de tal forma que:

$$F_x = S3 + S4 - S1 - S2$$

Para el cálculo de F_y , los sensores situados en la parte inferior se consideran valores negativos de y, y los situados en la parte superior se consideran positivos, de tal forma que:

$$F_y = S2 + S3 - S1 - S4$$

Los valores obtenidos de F_x y F_y son los que se mostraran en el grafico XY. Pero, antes de ello será necesario realizar el cálculo del offset de cada sensor sin encontrarse ninguna persona encima de la plataforma, ya que, aunque el ajuste de cero de los sensores ya se realizo por hardware, el comportamiento térmico de las resistencias y las galgas puede provocar este offset.

A continuación se describen los procedimientos utilizados para la elaboración de la aplicación.

Primeramente se inicializan las variables a utilizar durante la aplicación además de la configuración del puerto serie, donde se definirá el puerto COM a utilizar, número de bits, etc. Si la configuración es correcta y se ha establecido la comunicación entre los dispositivos, se mostrara un mensaje en la pantalla indicando: COM OK. En caso contrario se mostrará el mensaje: Error en COM.

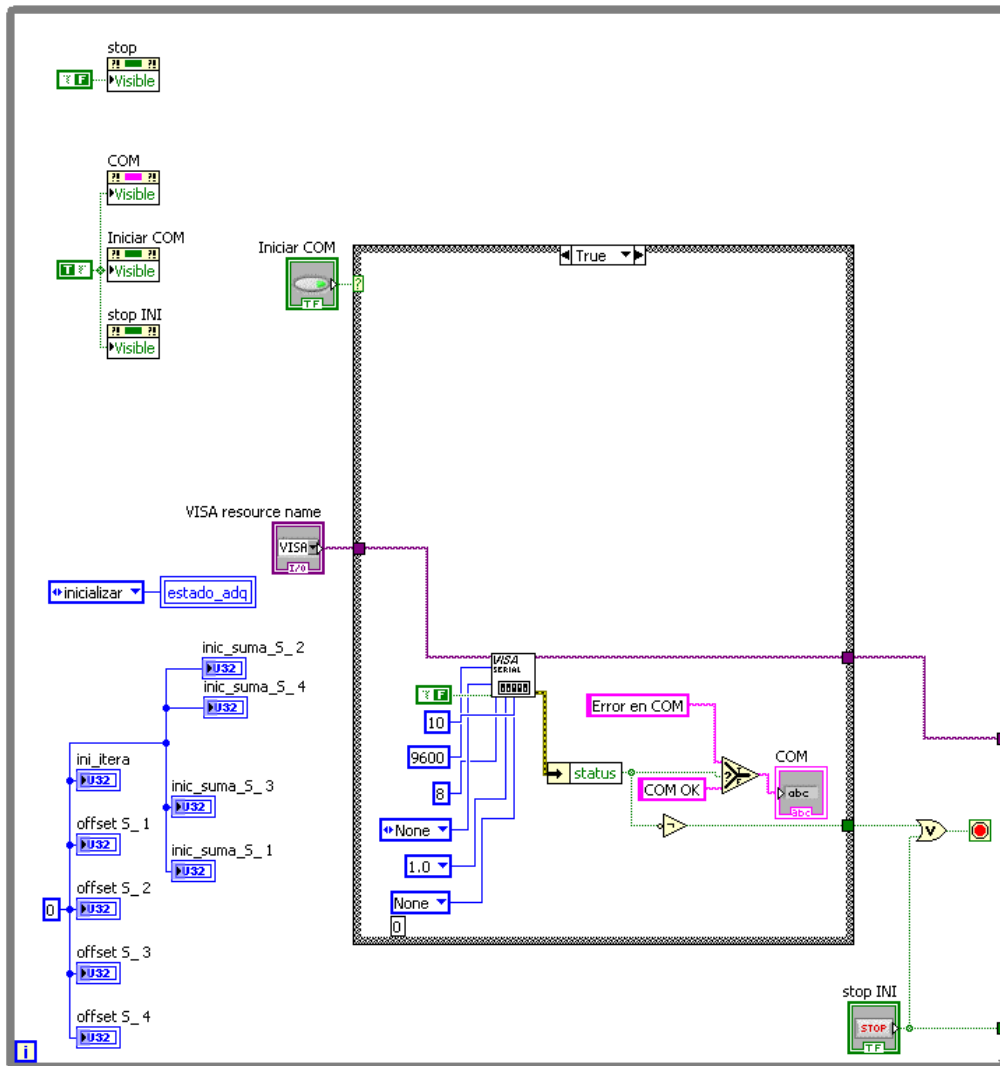


Figura 43 Bucle while de inicialización

En el siguiente bucle, mediante la función Read Visa se leerán los bytes del puerto serie. En este momento se empieza el tratamiento de los datos, primero separando los valores de cada sensor utilizando la función Search/Split String, que permite buscar un carácter en una cadena y separar la cantidad especificada de valores que le siguen. en nuestro caso, esta función es muy útil para separar los valores de S1, S2, S3 y S4, introduciendo como carácter de búsqueda S1, S2, S3 o S4 y especificando que separe los 4 valores que le siguen. Una vez separados los valores de los 4 sensores se efectúa un pequeño filtrado obteniendo la mediana de los valores de cada sensor, éstos se introducirán en una estructura case, que contará con 2 posibles casos, la inicialización y la adquisición. Inicialmente se realizara la inicialización, que corresponde al cálculo del offset de cada sensor. En la cual se adquirirán 100 muestras de cada sensor (sin ninguna persona sobre la plataforma) y se hará la media una vez realizado esto se activara la adquisición. En la adquisición, el valor obtenido será el offset que se deberá restar al valor adquirido del sensor con la persona sobre la plataforma, también se procederá a calcular y mostrar por pantalla la posición del centro de gravedad aplicando las formulas explicadas anteriormente utilizando la función Formula Node, que nos permite utilizar formulas matemáticas y expresiones similares al C, en ella se introducirán los datos como se muestra en la Figura 44.

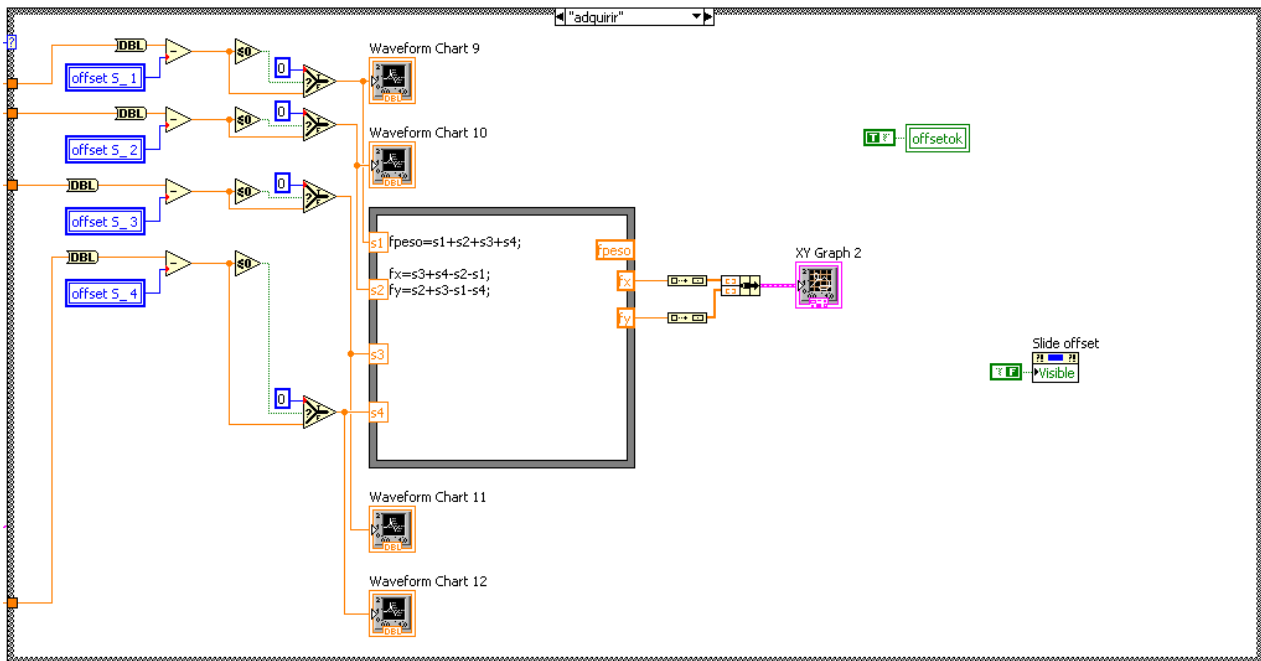


Figura 44 Funciones empleadas en el caso Adquirir

En la Figura 45 se muestra la vista del panel frontal, en él se puede seleccionar el puerto COM utilizado para la comunicación, y visualizar los datos del buffer, los valores de los 4 sensores, la grafica de posición del centro de gravedad, y las graficas de las señales de cada sensor sin filtrado, con filtrado y sin offset. El LED offsetok encendido indica que el cálculo del offset ha finalizado y que por lo tanto la persona ya puede subir a la plataforma.

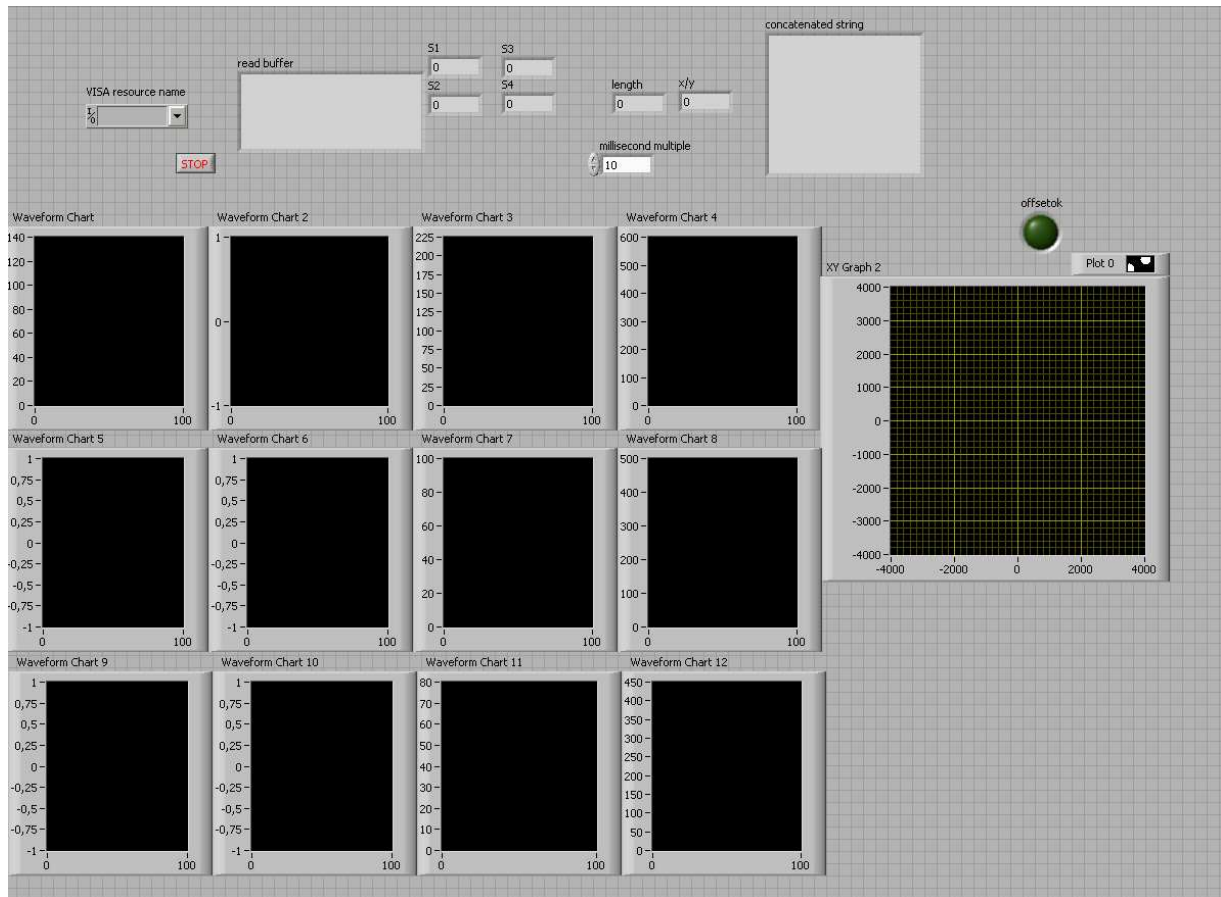


Figura 45 Vista del panel frontal

7.Presupuesto

En este capítulo se incluye el presupuesto completo del proyecto

7. Presupuesto

A continuación se muestra el presupuesto del proyecto distinguiendo entre los costes de los componentes necesarios para realizar la placa de circuito impreso, el coste de la plataforma y el coste total del proyecto, no se incluyen gastos de ingeniería ni de montaje y los precios indicados corresponden al precio de venta al público, no al por mayor:

Nombre	Descripción	Cantidad	Costo por unidad	Total
Componentes				
Resistencia 1K Ω	Resistencia 5% 1/4W	8	0,020 €	0,160 €
Resistencia 10K Ω	Resistencia 5% 1/4W	1	0,020 €	0,020 €
Resistencia 820 Ω	Resistencia 5% 1/4W	1	0,020 €	0,020 €
Resistencia 10 Ω	Resistencia 5% 1/4W	1	0,020 €	0,020 €
Potenciómetro 500 Ω	Potenciómetro 23 vueltas T93YB	4	1,300 €	5,20 €
Potenciómetro 100 Ω	Potenciómetro 23 vueltas T93YB	4	1,300 €	5,20 €
Regleta 3	Bloque terminal 3 vías	4	0,770 €	3,08 €
Regleta 2	Bloque terminal 2 vías	1	0,540 €	0,54 €
Tira de pines	Tira 40 pines Macho	1	1,100 €	1,100 €
Tira de pines	Tira 40 pines Hembra	1	1,400 €	1,400 €
Condensador 100nF	Condensador cerámico 63V	1	0,030 €	0,030 €
Condensador 100nF	Condensador de Tántalo 35V	4	0,920 €	3,680 €
Condensador 4,7 μ F	Condensador electrolítico 35V	3	0,047 €	0,141 €
Condensador 0,33 μ F	Condensador de Tántalo 35V	2	0,550 €	1,100 €
Condensador 1 μ F	Condensador de Tántalo 35V	2	0,140 €	0,280 €
Condensador 10 μ F	Condensador de Tántalo 25V	2	0,400 €	0,800 €
LED	LED rojo estándar	1	0,085 €	0,085 €
Diodo 1N4148	Diodo pequeña señal	1	0,074 €	0,074 €
AD620	Amplificador de instrumentación. Package: DIP8	4	8,300 €	33,200 €
dsPIC33FJ32GP204	Microcontrolador dsPIC. Package: TQFP	1	4,180 €	4,180 €
Modulo Promi-ESD02	Modulo Bluetooth clase 2	1	54,000 €	54,000 €
LM7805	Regulador tensión 5V. Package: TO-220	1	0,580 €	0,580 €
LM7905	Regulador tensión -5V. Package: TO-220	1	0,530 €	0,530 €
TC7660	Convertidor tensión. Package: DIP8	1	0,630 €	0,630 €
TC1262	Regulador tensión 3.3V. Package: TO-220	1	1,240 €	1,240 €
Conector pila	Conector de batería, PP3	1	0,250 €	0,250 €
Zócalo DIP8	Zócalo 8 contactos	5	0,105 €	0,525 €
Placa cobre doble cara		1		
Total de componentes				118,07 €
Estructura				
Bascula SilverCrest	Bascula de Baño digital	1	24,95 €	24,95 €
Total de Estructura				24,95 €
Total del Proyecto				143,02 €

Tabla 5 Presupuesto del proyecto

8. Conclusiones

En este capítulo se definen las conclusiones obtenidas de la elaboración del proyecto y se establecen posibles líneas futuras de trabajo

8. Conclusiones

Se ha conseguido diseñar un sistema de posturografía de bajo coste, portátil y totalmente autónomo, que permite obtener señales proporcionales a la fuerza aplicada en cada uno de los sensores de la plataforma. También se ha logrado establecer una comunicación inalámbrica mediante Bluetooth con el ordenador para la transmisión de las tramas de datos elaboradas en el microcontrolador.

El resultado es un sistema muy versátil que se puede utilizar en gran cantidad de dispositivos como pueden ser ordenadores, teléfonos móviles, PDA, tabletas, etc., utilizando cualquier sistema operativo que permita trabajar con puerto serie virtual por bluetooth como pueden Windows, Linux o Android.

En este proyecto se han aplicado conocimientos adquiridos durante la carrera en asignaturas como Instrumentación Industrial y Laboratorio de Instrumentación Industrial para el acondicionamiento de las señales, o Microcomputadores e Informática Industrial para el procesado, creación de tramas y transmisión de los datos en el microcontrolador

También se han aprendido nuevos conocimientos como el diseño de circuitos impresos y el fresado con ProtoMat, así como el manejo del software correspondiente al diseño de la PCB (Altium Designer Summer), a la programación del microcontrolador (compilador C30 de Microchip mediante el entorno MPLAB), a la configuración del Bluetooth (PromiWin) y a la elaboración de la aplicación de ejemplo. Además, se han ampliado los conocimientos en el ámbito de los microcontroladores.

8.1. Líneas futuras de trabajo

La versatilidad del sistema abre muchas líneas futuras de trabajo.

La incorporación del microcontrolador dsPIC con avanzadas prestaciones permite en un futuro ampliar el sistema para hacerlo mas autónomo incluyendo por ejemplo una pantalla en la estructura que permita la visualización directa de los datos mediante el tratamiento y análisis de las señales en el propio microcontrolador.

Además de la evaluación del equilibrio, otra posible ampliación del proyecto sería la evaluación del paso en el paciente, modificando la estructura del sistema y añadiendo los cálculos necesarios para ello.

También se puede trabajar en la mejora de la aplicación del ordenador para crear una interfaz con mayor usabilidad.

9. Bibliografía

En este capítulo se exponen las referencias tanto de libros, artículos, manuales como de páginas web empleadas para el desarrollo del proyecto.

9. Bibliografía

Libros:

Angulo, J. M.; Angulo, I.; García, B. y Vicente, J.; “MICROCONTROLADORES AVANZADOS dsPIC. Controladores digitales de señales. Arquitectura, programación y aplicaciones” Ediciones Paraninfo, 2005.

Pallás Areny, R.; “Sensores y acondicionadores de señal” Editorial Marcombo, 2003.

Pérez, M. A.; Álvarez, J. C.; Campo, J.C.; Ferrero, F. J. y Grillo, G. J.; “Instrumentación Electrónica” Editorial Thompson, 2003.

Peña, M. A. y Cela, J. M.; “Introducción a la programación en C” Ediciones UPC, 2000.

Artículos:

Ortuño, M. A.; Martín, E. y Barona, R.; *Posturografía estática frente a pruebas clínicas en ancianos con vestibulopatía* Acta Otorrinolaringológica Española, España 2008.

Ronda, J.M.; Galvañ, B.; Monerris, E. y Ballester, F.; *Asociación entre síntomas clínicos y resultados de la posturografía dinámica* Acta Otorrinolaringológica Española, España, 2002.

Manuales:

Microchip *dsPIC 33F Family Reference Manual*. Disponible:

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2556

ICD 2 *In-Circuit Debugger User's Guide*. Disponible:

<http://ww1.microchip.com/downloads/en/DeviceDoc/51331C.pdf>

Microchip *MPLAB® IDE User's Guide*. Disponible:

http://ww1.microchip.com/downloads/en/DeviceDoc/MPLAB_User_Guide_51519c.pdf

Páginas Web:

Consulta de conceptos generales:

<http://www.wikipedia.org/>

Información Posturografía:

<http://www.epsevg.upc.edu/cetpd/>

<http://resourcesonbalance.com/neurocom/>

<http://www.cun.es/la-clinica/servicios-medicos/otorrinolaringologia/mas-sobre-el-departamento/unidades/posturografia-area-de-patologia-vestibular/>

[Consultado el 18/12/2010]

Ayuda en el diseño PCB:

<http://www.altium.com/>

Conceptos sobre código, configuración y especificaciones del microcontrolador:

<http://www.microchip.com/>

<http://c.conclase.net/>

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002

Información Presupuesto:

<http://es.farnell.com/>

<http://www.diotronic.com/>

Anexo I

En este anexo se incluye un tutorial para el diseño de PCB utilizando el software Altium Designer Summer.

Anexo I: Diseño de una PCB en Altium Designer

Altium designer summer es un software para el diseño de sistemas electrónicos, como son el diseño de placas de circuito impreso (PCB), de software embebido o FPGA, y que comprende el diseño de esquemáticos y PCB, simulación, creación de librerías, entre muchos otros.

A continuación se detallaran los pasos a seguir para la consecución de un diseño completo en PCB, y que comprende los siguientes pasos:

- Creación de un proyecto
- Diseño del Esquemático
- Creación de librerías
- Diseño de la PCB

A1.1. Creación de un proyecto

El proyecto es la base y el punto de partida para la elaboración de cualquier diseño y que contendrá el conjunto de documentos de diseño cuyo resultado final es una única implementación.

Existen distintos tipos de proyectos dependiendo del resultado final:

- **PCB Project:** Comprenderá el conjunto de documentos de diseño necesarios para la elaboración de una placa de circuito impreso.
- **FPGA Project:** Comprenderá el conjunto de documentos de diseño que pueden ser procesados para la programación de un FPGA
- **Embedded Project:** Comprenderá el conjunto de documentos de diseño necesarios para la elaboración de una aplicación software que puede ser embebida por el procesador en un producto electrónico
- **Core Project:** Comprenderá el conjunto de documentos de diseño necesarios para la elaboración de una representación EDIF (modelo) de un componente funcional que puede ser implementado en un FPGA.
- **Integrated Library:** Comprenderá el conjunto de documentos de diseño necesarios para la elaboración de una Librería integrada.
- **Script Project:** Comprenderá el conjunto de documentos de diseño que contendrán uno o más scripts de Altium Designer.

Como hemos dicho anteriormente, nuestro objetivo es la elaboración de una placa de circuito impreso, por lo que seleccionaremos PCB Project en: File>>>New>>>Project>>PCB Project

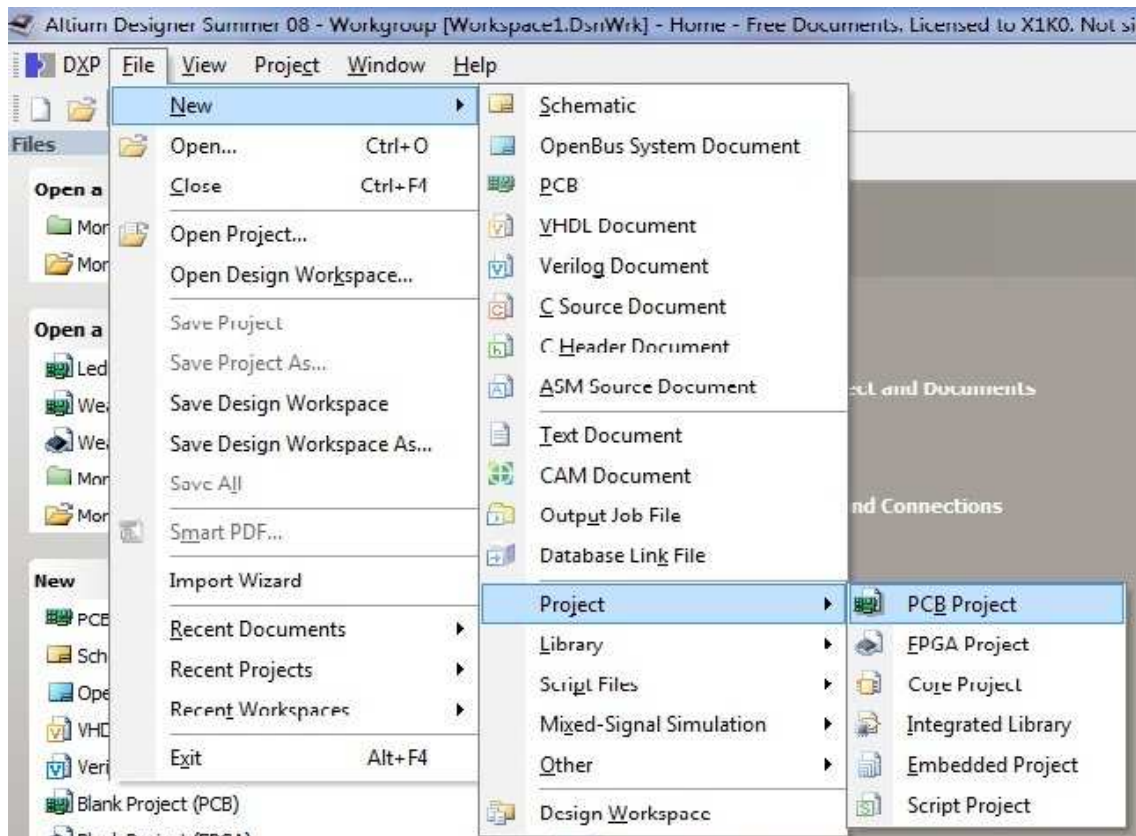


Figura 46 Creación de un PCB Project

En la ventana Projects (lateral izquierdo) nos aparecerá:

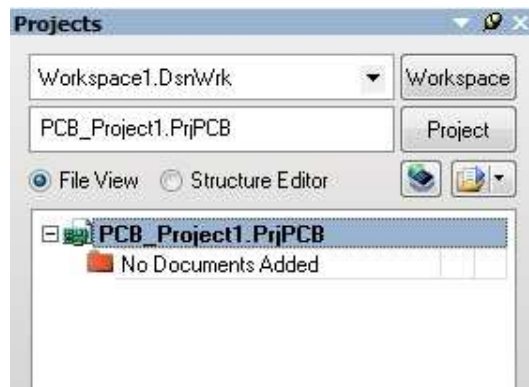


Figura 47

Por defecto el proyecto tiene el nombre PCB_Poject1.PrjPCB, para cambiarlo hacemos clic derecho sobre él y seleccionamos Save Project As..., seleccionamos la ubicación, le asignamos un nombre (en este ejemplo PlacaPFC) y hacemos clic en Guardar.

A continuación se procederá a añadir los documentos de diseño necesarios para elaborar una PCB, y que son el esquemático y el PCB.

Por lo tanto el PCB Project contendrá estos dos documentos, el esquemático, que estará formado por los símbolos de los componentes cableados como corresponda, y el PCB en el cual se transfiere el diseño del esquemático en forma de footprints (dimensiones físicas de cada componente) y se unen mediante pistas.

Para añadir estos documentos al proyecto, hacemos clic derecho sobre él y seleccionamos: Add New to Project>>Schematic y Add New to Project>>PCB.

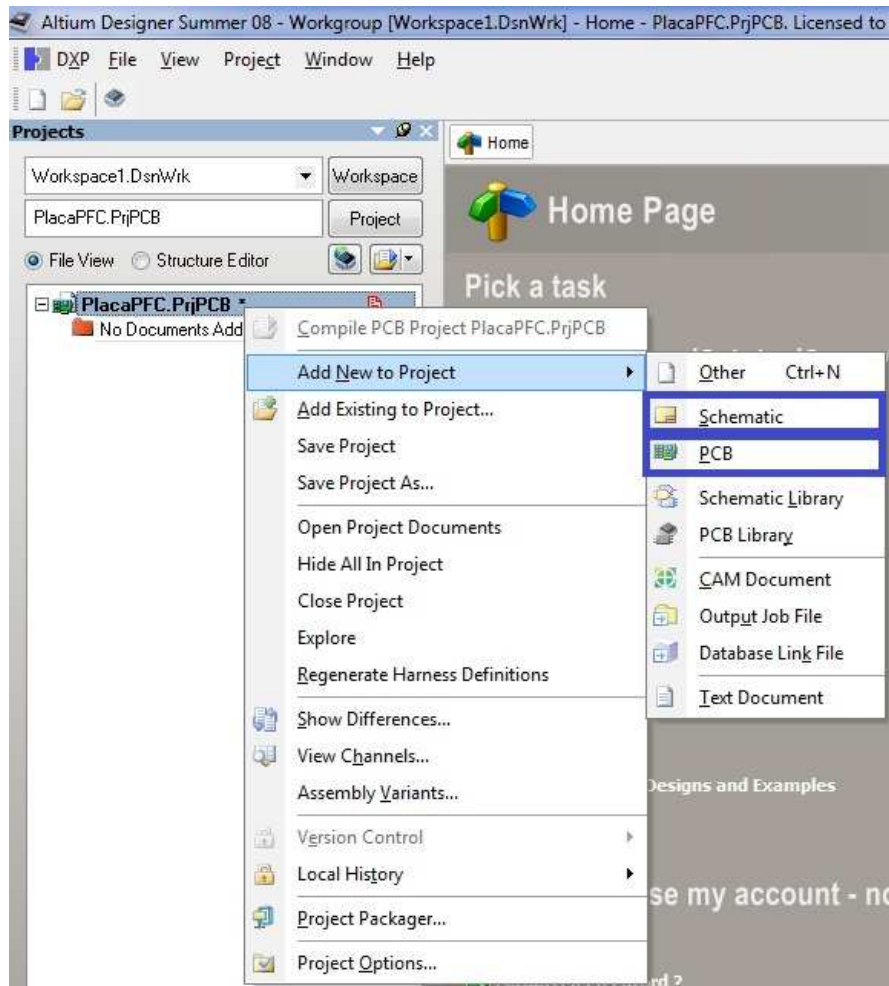


Figura 48

A continuación debemos asegurarnos de que ambos documentos se guardan en la misma ubicación que el proyecto haciendo clic derecho sobre cada uno de ellos y seleccionando Save As...

Ahora ya podemos proceder al diseño del esquemático.

A1.2. Diseño del esquemático

Antes de empezar con la colocación de los símbolos de los componentes hay que ajustar las opciones para el esquemático, como son las dimensiones de la hoja de trabajo, las cuadrículas las unidades...

Primero de todo haremos clic izquierdo sobre el esquemático que hemos creado en la ventana Projects. Luego hacemos clic en Design>>Document Options, se abrirá una ventana como la que se muestra a continuación:

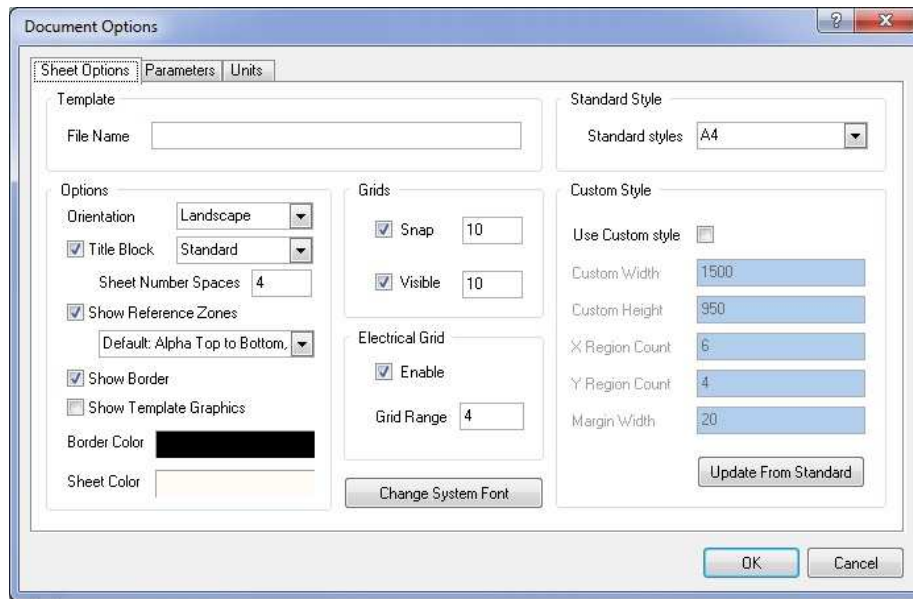


Figura 49

En la pestaña Sheet Options podemos modificar propiedades de la hoja de trabajo, como es la orientación en Orientation (Landscape: Horizontal; Portrait: Vertical), las dimensiones en Standard Styles (A4, A3, A2...) o las cuadrículas en Grids (Snap: Cuadrícula para la colocación de los símbolos; Visible: Cuadrícula para la navegación) y Electrical Grids (Cuadrícula para la ayuda en las conexiones).

En la pestaña Units podemos escoger las unidades de medida con las que trabajaremos entre Sistema Imperial y Sistema métrico. Escogeremos Sistema Métrico y las unidades Milímetros:

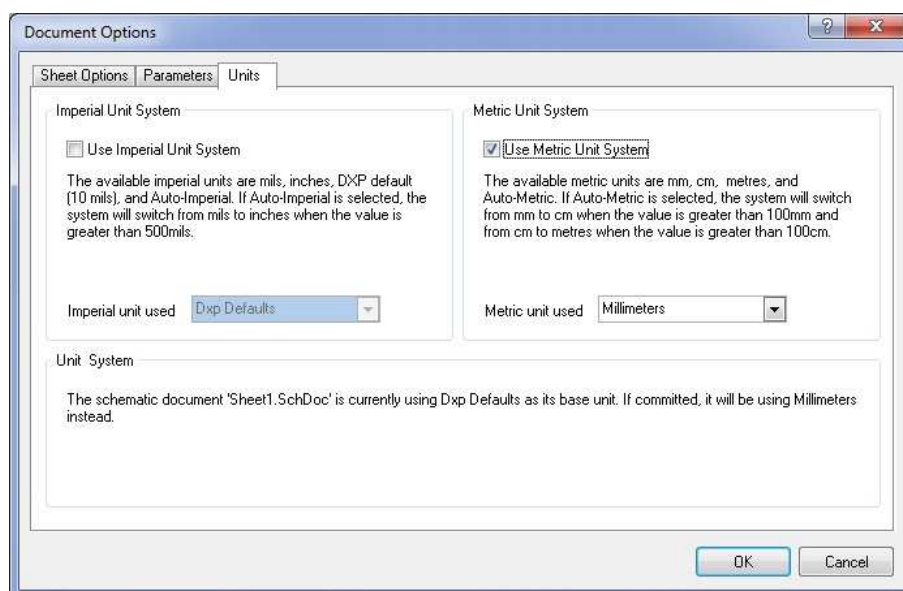



Figura 50

Una vez ajustadas las opciones ya podemos proceder a realizar el diseño del esquemático.

Para insertar el símbolo de un componente de una librería seleccionaremos Place>>Part, o clicando sobre el icono  de la barra de herramientas en la parte superior. Se abrirá una ventana como la que se muestra a continuación y haremos clic sobre el botón ... para acceder a las librerías de símbolos:

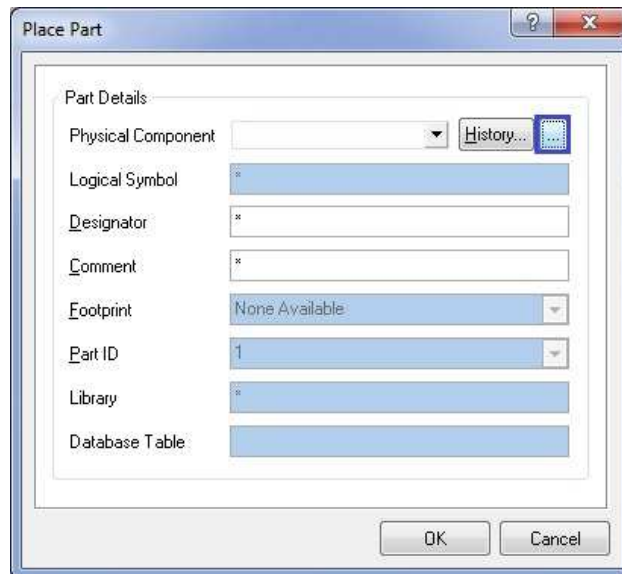


Figura 51

Se abrirá una nueva ventana como la que se muestra en la Figura 52, en el apartado Libraries podemos acceder a las principales librerías como son Miscellaneous Devices y Miscellaneous Connectors. En Miscellaneous Devices podemos encontrar componentes de uso general como son resistencias, potenciómetros, condensadores, transistores, diodos, etc. En Miscellaneous Connectors podemos encontrar conectores de uso general como son conectores DB-9, coaxiales, tiras de pines, etc.

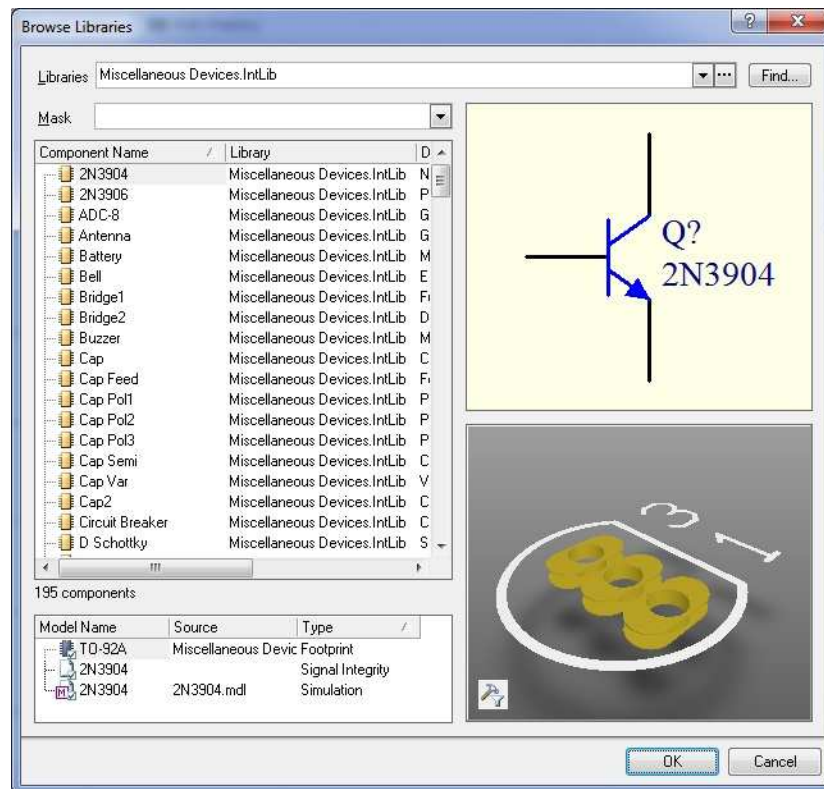
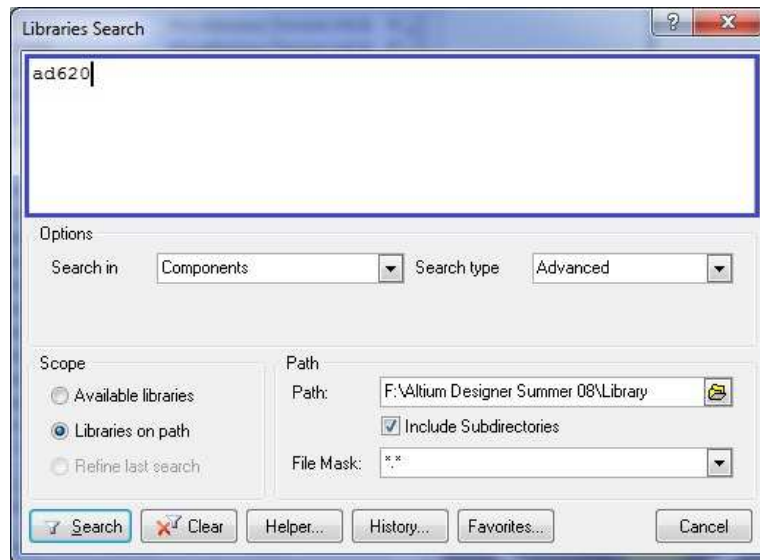


Figura 52

Si nos interesa un componente más específico de un fabricante en concreto podemos buscar dicho componente en las librerías de los distintos fabricantes haciendo clic en el botón Find... de la misma ventana.

Se abrirá una nueva ventana como la de la Figura 53. En la zona que hemos enmarcado de color azul se introduce el nombre del componente a buscar (por ejemplo: ad620), y los demás campos se configuraran como se muestra en la figura 8 para que la búsqueda se realice en las librerías disponibles. En el campo Path seleccionaremos la subcarpeta Library de la ubicación donde se encuentra instalado Altium Designer. Una vez configurados los criterios de búsqueda haremos clic en Search.

**Figura 53**

Una vez finalizada la búsqueda seleccionamos el componente que nos interesa así como el footprint (parte inferior izquierda de la ventana) y hacemos clic en OK.

Si la librería no está instalada nos aparecerá el siguiente mensaje:

**Figura 54**

Hacemos clic en Yes para instalar la librería.

Se abrirá la siguiente ventana correspondiente al componente:

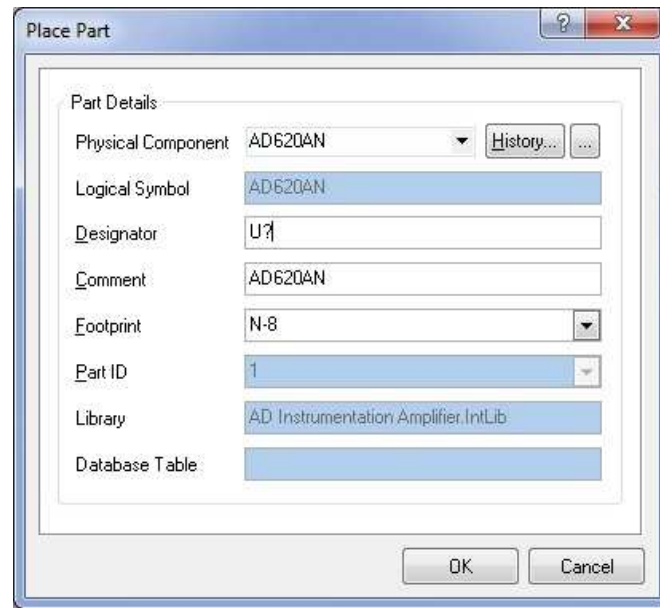


Figura 55

Es importante asignar un número en designator (por ejemplo U0), y que no se repita ningún designator en ningún componente. Posteriormente hacemos clic en OK.

Aparecerá el símbolo del componente sobre la hoja de trabajo. Lo moveremos con el movimiento del ratón para colocarlo donde deseemos. Si queremos rotar el componente pulsaremos la tecla Space. Si queremos editar las propiedades del componente pulsaremos la tecla Tab. Para fijar el símbolo en la hoja de trabajo hacemos clic izquierdo en la ubicación que nos interese en la hoja.


Otro icono importante y útil de la barra de herramientas es el correspondiente a las fuentes de alimentación



, en el cual, al clicar sobre la flecha se mostrara un desplegable como el que se muestra en la Figura 56 con los diferentes modos de alimentación para el circuito. Permite definir una red de alimentación o masa en cualquier lugar del diseño.



Figura 56

Por ejemplo, para la masa del circuito, en vez de tener que unir todos los puntos que van a masa en el esquemático, utilizaremos el símbolo  en todos los puntos del circuito que vayan conectados a masa, y evitaremos así tener que hacer excesivas conexiones que hacen engorroso el esquemático.

Esta es la rutina a seguir para la ubicación de los símbolos componentes y fuentes de alimentación en la hoja de trabajo. Una vez colocados todos los símbolos procederemos a conectarlos, para ello podemos escoger entre cable, bus, etc.


A1.3. Creación de librerías

Aunque Altium dispone de una gran variedad de librerías de diferentes fabricantes, es posible que algún componente de nuestro circuito no se encuentre en ellas, o que las características del footprint no se correspondan con las características físicas de nuestro componente. Es por ello que en algunas ocasiones deberemos crear nosotros mismos el símbolo y el footprint de un componente. A continuación se detallan los pasos a seguir para crear una librería.

Como explicamos en el apartado “A.1 Creación de proyectos” el proyecto es la base y el punto de partida para todo diseño y en función de nuestro diseño escogeremos un tipo de proyecto. Para la creación de proyectos se utiliza Integrated Library, el cual comprenderá el conjunto de documentos de diseño necesarios para la elaboración de una Librería integrada y los cuales serán un esquemático para el diseño del modelo y un PCB para el diseño del footprint. Para ello haremos clic en File>>New>>Project>>Integrated Library. En la ventana Projects (lateral izquierdo) nos aparecerá el nuevo proyecto, el cual guardaremos en una carpeta de la subcarpeta Library que se encuentra en la ubicación donde está instalado Altium Designer, haciendo clic derecho sobre él y seleccionando Save Project As.... A continuación añadiremos el esquemático y el PCB haciendo clic derecho otra vez sobre el proyecto y seleccionando Add New to Project>>Schematic Library y Add New to Project>>PCB Library. Guardaremos el esquemático en la misma ubicación y el pcb en la subcarpeta de Library llamada Pcb, que es donde se encuentran todos los footprints.

A continuación se procederá al diseño del símbolo del componente mediante el documento esquemático. Nos situamos en la hoja de trabajo, y lo primero que hay que hacer es nombrar el componente haciendo clic en Tools>>Rename Component, se abrirá una ventana en la que introduciremos el nombre del componente.

Haciendo clic en Place o en el icono desplegable  de la barra de herramientas podemos seleccionar entre diversas formas geométricas para crear el modelo del componente (rectángulos, arcos, líneas), además

de los pines que precise el componente seleccionando  del desplegable. Haciendo doble clic sobre los pines podremos modificar sus características, como por ejemplo el nombre, el número del designator (es importante que el número del designator coincida con el número del pad que le corresponda en el footprint o pcb), la longitud del pin, etc...

Una vez dibujado el modelo y colocados los pines se procederá al dibujo del footprint en el documento Pcb que hemos creado anteriormente.

Para el diseño del footprint tenemos dos opciones:

- Utilizar el Component Wizard cuando se trate de componentes con encapsulado tipo BGA, DIP, LCC, PGA, QUAD, SOP, SBGA, SPGA o resistencias, condensadores y diodos.
- Hacer el diseño del footprint manualmente cuando se trata de componentes con características concretas con encapsulados no mencionados anteriormente.

Component Wizard

Cuando se trata de encapsulados como los descritos anteriormente, mediante el Component Wizard se puede hacer el diseño del footprint de forma fácil y sencilla.

Por ejemplo, para el footprint de un componente con encapsulado DIP8:

Seleccionamos el documento PcbLib, en el cual se hará el diseño del footprint. Seleccionamos Tools>>Component Wizard. Se abre una ventana de inicio del asistente como se muestra a continuación:



Figura 57 Pantalla de inicio del Component Wizard

Navegaremos a través del asistente mediante los botones **Back**, **Next** y **Finish**.

Hacemos clic en **Next** para acceder a la siguiente ventana, en la cual se mostraran los encapsulados a elegir y las unidades con que se trabajará para definir las medidas del componente. Para este ejemplo se selecciona **Dual In-line Packages (DIP)** como encapsulado y **Metric (mm)** como unidad.

Hacemos clic en **Next** y accedemos a la pantalla para el diseño de los pads, que son las zonas reservadas para la correcta soldadura de cada uno de los pines del componente. En esta página se introduce el diámetro del agujero y las dimensiones en el eje x y el eje y del componente. Las dimensiones del componente se obtienen consultado el datasheet del componente. Para las dimensiones de los pads es importante tener en cuenta que el pad debe rodear el componente de manera que permita una buena soldadura, sobre todo en los integrados las dimensiones del pad suelen ser el doble en un eje que en el otro. En este caso los valores serán los siguientes:

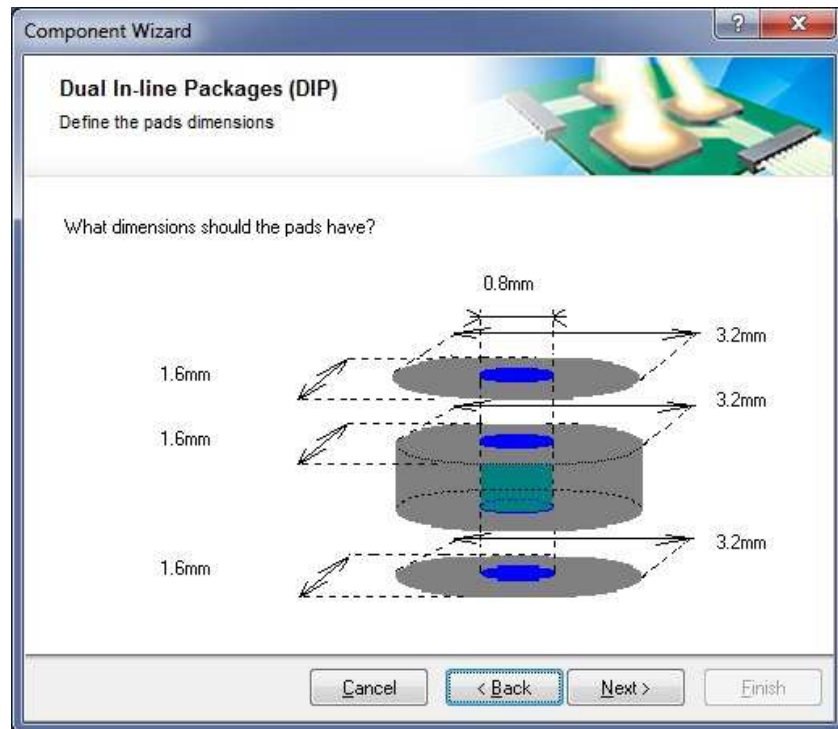


Figura 58 Dimensionado de los pads en el Component Wizard

Hacemos clic en **Next** y en la siguiente pantalla nos aparecerá se procederá a introducir la distancia en el eje x y el eje y entre los pads, consultando el datasheet de componente introducimos:

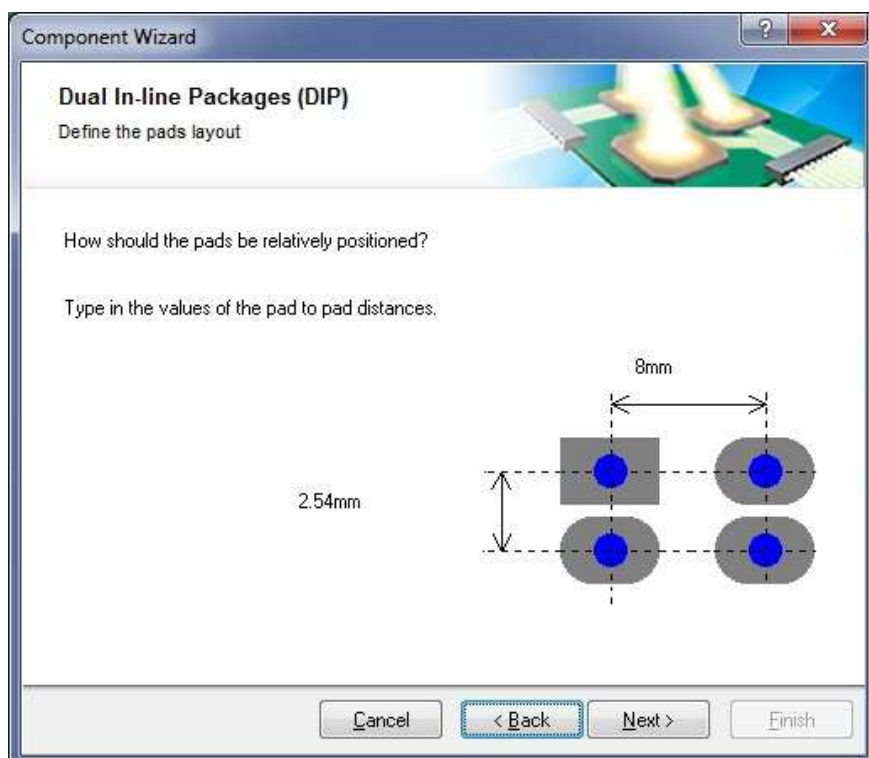


Figura 59 Dimensionado de la distancia entre los pads

Hacemos clic en **Next** y aparecerá una ventana para introducir el grosor de la línea que define el perímetro del componente. Esta línea únicamente nos interesa para hacernos una idea del espacio que ocupa el componente en la board. Dejamos el valor por defecto y hacemos clic en **Next**.

En la siguiente ventana se debe introducir el número de pads para el componente, que serán tantos como pines tenga. En este caso, tratándose de un encapsulado DIP8, introducimos 8, cuatro en cada lado tal y como muestra la Figura 60:

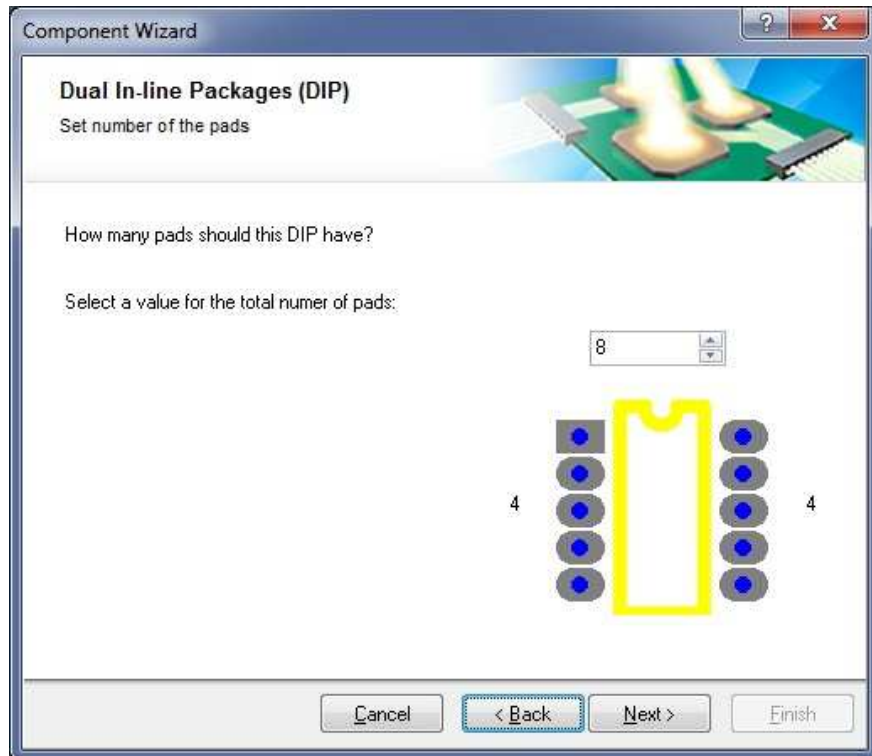





Figura 60 Número de pads en el Component Wizard

Hacemos clic en **Next** y en la siguiente ventana se deberá introducir el nombre del componente, en este caso le llamaremos DIP8.

Una vez introducido el nombre hacemos clic en **Next** y la siguiente ventana nos indicara que el asistente ya ha finalizado. Hacemos clic en **Finish** y nos aparecerá el footprint en nuestro documento PcbLib. Lo guardamos y ya podremos proceder a añadir el footprint al esquemático.

Diseño manual del footprint

Cuando se trata de componentes con un encapsulado especial deberemos hacer el diseño manualmente. Para ello se utilizan las herramientas seleccionando Place o en el menú de herramientas. Las más utilizadas son:

- **Line** : Seleccionaremos esta herramienta para dibujar las líneas que definen el perímetro de componente, normalmente se utiliza la capa Top Overlay (parte inferior del área de trabajo). Haciendo doble clic sobre una línea ya dibujada se muestran las características de ésta y nos permite modificar el grosor, la ubicación en los ejes XY y su longitud
- **Pad** : Seleccionaremos esta herramienta para colocar los pads del componente. Debe tener el mismo número de pads como pines hay en el esquemático y corresponderse el designator del pad con el del pin al que corresponda. Haciendo doble clic sobre un pad podemos editar sus características como son el designator, el diámetro del agujero y las dimensiones y forma del pad.
- **Arc, Circle** : Seleccionaremos esta herramienta para dibujar arcos o círculos que definen el perímetro de componente, normalmente se utiliza la capa Top Overlay. Haciendo doble

clic sobre un arco o círculo ya dibujado se muestran las características de éste y nos permite modificar el grosor, la ubicación en los ejes XY el radio y el inicio y fin del ángulo.

Una vez diseñado el componente guardamos el documento PcbLib y ya podremos proceder a añadir el footprint al esquemático.

Para añadir el esquemático al footprint nos situamos en el documento esquemático seleccionándolo en la ventana projects. En la parte inferior del documento hacemos clic sobre el botón **Add Footprint**. Se desplegará la siguiente ventana:

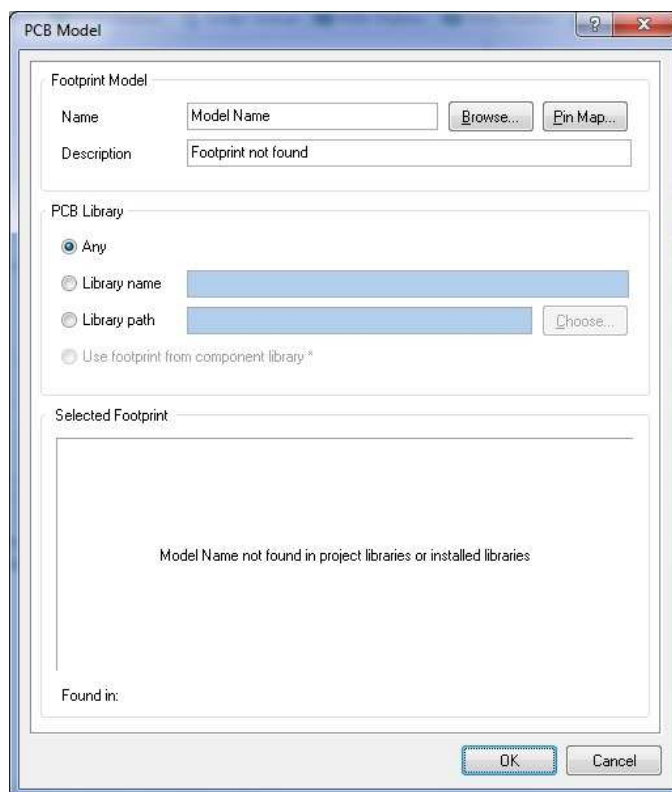


Figura 61 Ventana PCB Model

Hacemos clic en Browse para buscar el documento Pcb. Se abrirá una ventana que nos permitirá buscarlo en las librerías, como hemos dicho anteriormente el documento PcbLib se guarda en la carpeta Pcb, si no encontráramos el componente hacemos clic en Find y introducimos el nombre del footprint (por ejemplo DIP8) y en path seleccionamos la ubicación de la carpeta Pcb. Una vez encontrado hacemos clic en OK y veremos cómo se muestra el footprint añadido en la parte inferior del esquemático. Guardamos todos los documentos y hacemos clic con el botón derecho sobre el documento LibPkg y seleccionamos Compile Integrated Library. Guardamos el documento LibPkg y la librería ya estará lista para ser usada.

A1.4 Diseño de la PCB

Una vez creado el esquemático, abrimos el documento PCB que hemos creado antes hacemos clic en Design>>Import Changes From... Se abrirá una ventana que nos mostrará los componentes y las Nets a añadir, hacemos clic en Execute Changes y cerramos. Ahora ya vemos los footprints de los componentes en la hoja de trabajo sobre un fondo rojo. Hacemos clic sobre él y lo arrastramos hasta la zona destinada para la PCB, volvemos a hacer clic sobre ella y apretamos suprimir.

Ahora ya podemos organizar y ordenar los componentes haciendo clic sobre ellos los moveremos a donde queramos, y apretando la tecla SPACE del teclado haremos rotar el componente 90°.


Al arrastrar observamos que de los terminales del componente salen unas líneas, esto nos indica dónde va conectado el terminal.


También se puede utilizar el Autoplacer en Tools>>Component Placement>>Autoplacer.


Una vez ubicados los componentes en la placa empieza la tarea de conexonado mediante pistas. Para ello seleccionamos la capa en la que queremos que vaya la pista: Si únicamente se utiliza una cara se selecciona la capa Bottom Layer, si se utilizaran las dos capas se escogerá entre Bottom Layer o Top Layer.

Haciendo clic en Auto Route podemos seleccionar entre diferentes opciones para que el programa se encargue de hacer las pistas.

Aun así son mejores los resultados obtenidos haciendo las pistas manualmente. Las herramientas mas utilizadas para esta tarea son:

Route : Se utiliza para crear pistas. es importante que los cambios de dirección se realicen con ángulos de 35°. También hay que destacar que dos pistas de una misma capa no pueden cruzarse.

Vía : Se utiliza para pasar una pista de una capa a otra.

Polygon Plane : Se utiliza para insertar planos de masa. Al hacer clic se abre una ventana que nos permite configurar a que Net va conectado el plano de masa entre otras. Al hacer clic en OK, se cerrara la ventana y deberemos indicar las 4 aristas que forman el plano.

Una parte importante del diseño de la PCB son las Rules o reglas del proyecto: en ella se establecen los límites del grosor de las pistas, los pads, la distancia entre pistas, etc.

Anexo II

En este anexo se incluye un listado con los documentos y programas que se encuentran anexados en el formato electrónico

Anexo II. Documentos anexados en el formato digital

A continuación se listan los documentos anexados en el formato electrónico.

La carpeta Anexos está formada por las siguientes subcarpetas:

Aplicación LabView: Contiene los archivos con extensión .vi que permiten ejecutar la aplicación de ejemplo en LabView.

Código dsPIC: Contiene el proyecto elaborado en MPLAB con el código del programa del dsPIC

Datasheet y Manuales dsPIC: Contiene el datasheet del dsPIC y los manuales del ICD2 y del MPLAB

Manual Bluetooth: Contiene el manual del modulo bluetooth.